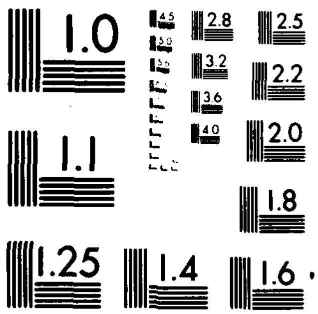
AD-A084 224 AERONAUTICAL SYSTEMS DIV WRIGHT-PATTERSON AFB OH DEPU--ETC F/6 17/9
TACTICAL TARGET ACQUISITION MODEL (TATAC). VOLUME II. USERS MAN--ETC(U)
DEC 77 S BAILY

UNCLASSIFIED ASD/XR-TR-79-5029-VOL-2 NL

END
DATE
FILMED
6-80
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

LEVEL III

12

ASD/XR-TR-79-5029

Ops / Analysis

Directorate Of Operations Research

ADA084224

TACTICAL TARGET ACQUISITION MODEL (TATAC)

VOLUME II

USERS' MANUAL

DECEMBER 1977

PREPARED FOR ASD BY

LULEJIAN AND ASSOCIATES, INC.

DEPUTY FOR DEVELOPMENT PLANNING
AERONAUTICAL SYSTEMS DIVISION
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

NOTICES

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights to permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Copies of this report should not be returned to the Aeronautical Systems Division unless return is required by security considerations, contractual obligations, or notice on a specific document.

Publication of this technical report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

This technical report has been reviewed and is approved for publication.

S. A. TREMAINE
Deputy for Development Planning

DISCLAIMER NOTICE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER (14) ASD/XR-TR-79-5029-VOL-2	2. GOVT ACCESSION NO. AD-A084 224	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) TACTICAL TARGET ACQUISITION MODEL (TATAC) [redacted] Users' Manual A034223		5. TYPE OF REPORT & PERIOD COVERED Volume II. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) Stewart/Baily		8. CONTRACT OR GRANT NUMBER(s) (12) 83
9. PERFORMING ORGANIZATION NAME AND ADDRESS DIRECTORATE OF OPERATIONS RESEARCH DEPUTY FOR DEVELOPMENT PLANNING AERONAUTICAL SYSTEMS DIVISION (AFSC)		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (16) A2354 63101F
11. CONTROLLING OFFICE NAME AND ADDRESS DIRECTORATE OF OPERATIONS RESEARCH (ASD/XROL) DEPUTY FOR DEVELOPMENT PLANNING WRIGHT-PATTERSON AFB, OH 45433		12. REPORT DATE (11) Dec[redacted]77 13. NUMBER OF PAGES 87
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) See Block 11.		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

See Block 16.

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Target Acquisition	Target Sensors
Tactical Air Power	Radar
Air-to-Ground Attack	Infrared
Atmospheric Effects	TV Systems
Tactical Scenarios	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A computer model for determining target acquisition capability of airborne sensors against ground targets is presented. The theory and general structure of the model plus details for the user are discussed. The model can be used to evaluate various types of electrooptical and radar sensors.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

403875

PREFACE

This is one of two volumes of a report describing a target acquisition model developed under contract to Lulejian and Associates by the Deputy for Development Planning (XRO), of the Aeronautical Systems Division. The model was developed to support in-house studies of tactical air-to-ground attack.

TABLE OF CONTENTS

SECTION		PAGE
I	INTRODUCTION	I-1
II	PROGRAM OPERATION	II-1
	A. EXECUTION LIST	II-3
	1. Determining the Run	II-5
	2. Data Entry	II-5
	3. Setting Up an Execution List	II-6
	B. OUTPUT	II-6
	1. List of Data Modifications	II-7
	2. Error Messages	II-7
	3. Standard Model Output	II-9
	4. Variations in Standard Model Output . . .	II-11
	C. PROGRAM LIBRARY DATA	II-12
	D. FIXED DATA	II-22
III	LOGICAL STRUCTURE	III-1
IV	FIXED DATA LISTING	IV-1
V	LIBRARY DATA LISTING	V-1
VI	SAMPLE PROBLEMS	VI-1
	A. VISUAL OBSERVER (VISOB) - SAMPLE PROBLEM A . .	VI-1
	B. FORWARD-LOOKING RADAR (FLR) - SAMPLE PROBLEM B	VI-4
	C. SYNTHETIC APERTURE RADAR (SAR) - SAMPLE PROBLEM C . . .	VI-7
VII	TATAC PROGRAM LISTINGS	VII-1

LIST OF ILLUSTRATIONS

FIGURE		PAGE
1	TATAC Deck Structure	II-1
2	TATAC Program Information Flow	II-2
3	Sample Execution List	II-4
4	Sample Computer Output	II-10
5	Execution List - Sample Problem A	VI-2
6	Computer Output - Sample Problem A	VI-3
7	Execution List - Sample Problem B	VI-5
8	Computer Output - Sample Problem B	VI-6
9	Execution List - Sample Problem C	VI-8
10	Computer Output - Sample Problem C	VI-9

LIST OF TABLES

1	LEGAL DATA NAMES	II-5
2	OPERATIONAL VARIABLES	II-13
3	TARGET DATA	II-14
4	BACKGROUND DATA	II-14
5	ENVIRONMENTAL/TERRAIN DATA	II-15
6	SEARCH DATA	II-15
7	VISUAL SENSOR DATA	II-16
8	FORWARD-LOOKING INFRARED	II-16
9	ACTIVE TV SENSOR DATA	II-17
10	PASSIVE TV	II-18
11	FORWARD-LOOKING RADAR, MTI	II-19
12	FORWARD-LOOKING RADAR, NON-MTI	II-20
13	SYNTHETIC APERTURE RADAR	II-21
14	INTEGERS IN FIXED DATA ARRAY IC	II-22
15	INTEGERS IN FIXED DATA ARRAY ICT	II-23
16	TABLES IN FIXED DATA ARRAY FDTAB	II-25
17	DATA IDENTIFICATION POINTERS	III-1
18	INTERNAL ARRAY NAMES	III-2

TACTICAL TARGET ACQUISITION MODEL (TATAC)

I. INTRODUCTION

This volume documents the operating procedures, input requirements, and output formats for the Tactical Target Acquisition (TATAC) computer model. After defining the problem this guide can be used to assist the user in setting up the necessary steps to make a computer run.

The computer program is comprised of the following sensor system models:

- Visual Observer (VISOB)
- Forward-Looking Infrared (FLIR)
- Television (TV)
  - Active (Illuminated)
  - Passive (Daylight)
- Forward-Looking Radar (FLR)
  - Moving Target Indicator (MTI) Mode
  - Non-MTI Mode
- Synthetic Aperture Radar

Section II of this guide describes the operation and structure of the program in terms of:

- Execution List (Input)
- Output
- Library Data
- Fixed Data

Section III describes the logical structure of the programs. Section IV contains a listing of the fixed data. Section V contains a listing of the library data. Section VI contains sample problems including the execution list and output from the computer run for each problem to demonstrate the method of using the model. Section VII contains a listing of all programs in the model.

II. PROGRAM OPERATION

This section describes the procedures necessary to run the TATAC computer model. Detailed in this section are:

- Description and form of the execution list;
- Form of output generated by a computer run;
- Library data description; and
- Fixed data description.

Figure 1 is a diagram of the program deck. Figure 2 summarizes the information flow in the program.

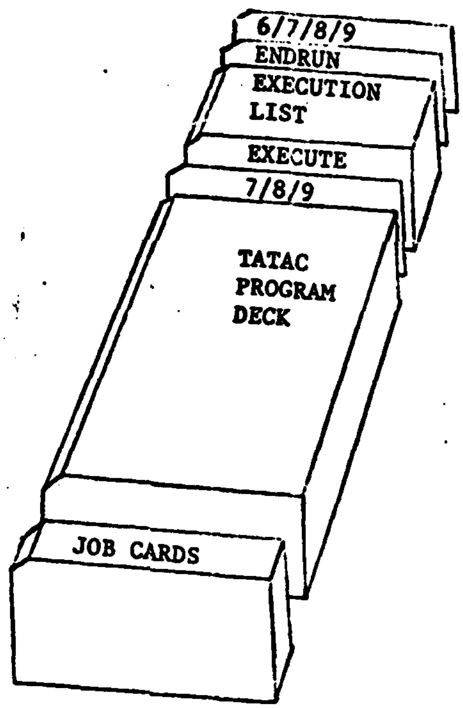


Figure 1. TATAC Deck Structure

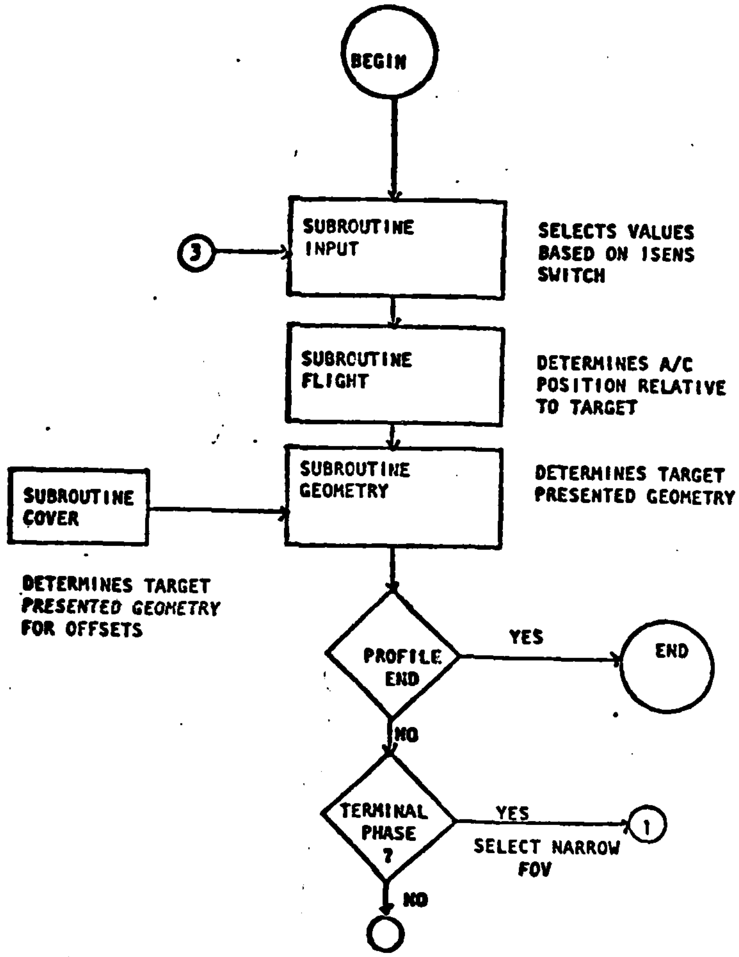


Figure 2. TATAC Program Information Flow

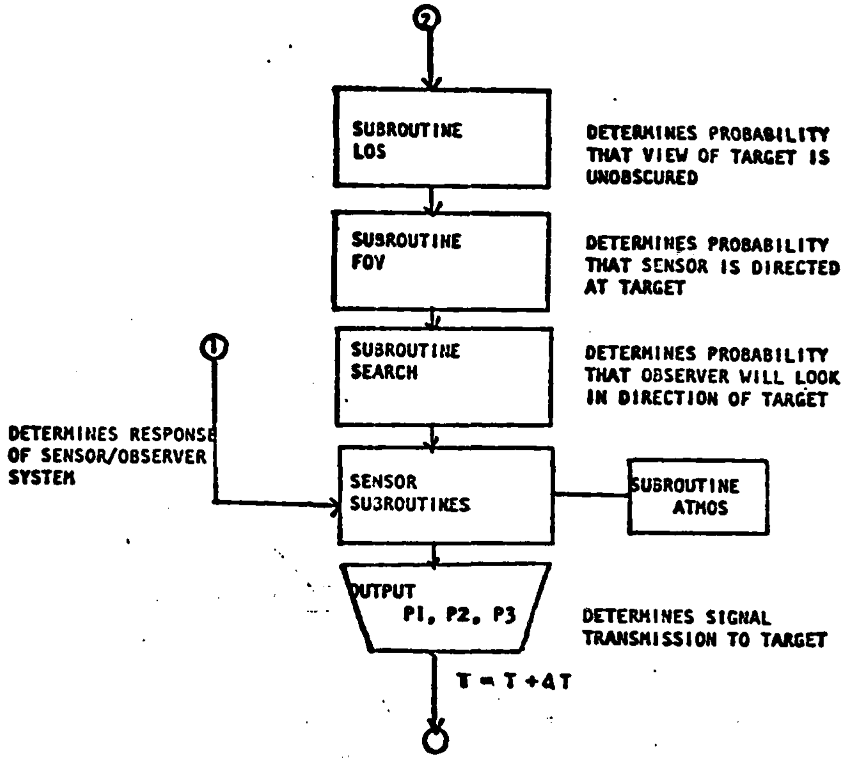


Figure 2. TATAC Program Information Flow (Continued)

A. EXECUTION LIST

The execution list follows the program deck as shown in Figure 1. It begins with an EXECUTE card and ends with an ENDRUN card. This list is the only portion of the data which must be formulated by the user before each computer run. As a reference to the following description, a sample execution list is shown in Figure 3.

The execution list performs the following functions in the model:

Determines the type of run to be executed; and

Denotes all data of the library to be modified.

These functions are described in the following paragraphs.

```
         8       16      24      32      40      48      56      64      72      80
 1 EXECUTE   2
 2 OPERATIONAL VAR             2
 3          3 .5000E+04
 4          4 .2000D+03
 5 ENDRUN
 6
 7
 8
 9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
```

Figure 3. Sample Execution List

1. Determining the Run

The first card in the execution list is the EXECUTE card (Figure 3, card 1). Besides indicating to the computer model that this is the beginning of the execution list, this card conveys the type of run to be executed. The following sensor runs may be executed:

Visual Observer (1);

Forward-Looking Infrared (2);

Active (illuminated) Television (3);

Passive (daylight) Television (4);

Forward-Looking Radar, MTI Mode (5);

Forward-Looking Radar, non-MTI Mode (6); and

Synthetic Aperture Radar (7).

To indicate the sensor run being set up, the user places one of the single-integer codes, shown in parenthesis in the above list, into column 11 of the EXECUTE card.

2. Data Entry

Besides the EXECUTE and ENDRUN cards the only other types of cards that may be needed are "data name" and "data modification" cards. A "data modification" card(s) is used in conjunction with a "data name" card. That is, a "data name" card need never be used unless a "data modification" card(s) immediately follows it.

a. Data Name Cards

The "data name" card (Figure 3, Card 2) has the legal name of the data being modified in columns 1-15 (left justified). Legal names for data used in the program are given in Table 1. The user then places an integer in columns 20-30 (right justified). This integer tells the input program how many modifications are to be made to that particular library entry. Then for each modification a "data modification" card follows. Recall that a "data name" card is only used if one or more modifications are made to that particular library "data name."

TABLE 1. LEGAL DATA NAMES *

OPERATIONAL VAR TARGET BACKGROUND	*	Only the first four characters are checked by the program. Thus, all data names in the list may be shortened.
ENVIRONMENTAL ** TERRAIN ** SEARCH SENSOR	**	Environmental and terrain data are combined into one array - either name may be used as a legal data name for this array.

b. Data Modification Cards

Each of these cards (Figure 3, cards 3 - 4) contains the number of the variable to be modified in columns 9 - 10 (right justified) and the new value assigned to it in columns 11 - 20 in E10.4 format (right justified). The variable number and its type are found in the library data tables in Subsection C of this Section. The number of "data modification" cards which follow a "data name" card must agree with the integer placed in columns 29 - 30 of the "data name" card.

3. Setting Up an Execution List

This subsection lists the steps which the user should follow when setting up an execution list to run the TATAC computer model.

Defining the problem:

(1) Define the problem to be run including the sensor system, operational variables, target, background, environmental/terrain, and search variables.

(2) Determine if the data needed is in the library. If not, "data name" and "data modification" cards for new entries will be required.

Filling out the execution list:

(3) List the EXECUTE card including the sensor number in column 11 for the run to be made.

(4) Insert all "data name" cards along with the proper number of "data modification" cards for each.

(5) List the ENDRUN card.

B. OUTPUT

There are basically three categories of output from the computer model:

List of data modifications;

Error messages; and

Standard model output.

Types of each output applicable to a specific run will be automatically generated by the program. A discussion of the categories

of output will be given in the next three subsections. Following these discussions will be a discussion on variations of standard model output among the sensor models. Reference will be made at that point to sample problems which illustrate the distinctions in output generated by computer runs.

1. List of Data Modifications

Preceding all standard output, a listing of data modifications made through the use of "data modification" cards in the execution list will be given. If no data modifications have been made, no output of this category will be generated. The list will include the legal data name for which modifications were made, the number of the variable modified, IVAR (found in the library data table for the applicable data name), and the value assigned to that variable. Space for output is reserved for as many as ten modifications per data name used. If more than ten "data modification" cards follow a "data name" card, a message is printed alerting the user to this occurrence. This in no way affects the program run. It simply means that all changes made after the tenth one, will not be listed on the output. There are two ways to avoid this situation. One is to repunch the entire portion of the library data array applicable to the library data name in question. The other is to distribute the "data modification" cards behind multiple "data name" cards containing the same data name. For example: if 15 data modifications are to be made to a library entry, set up a "data name" card with a "10" in columns 29 - 30, followed by ten "data modification" cards for the first 10 modifications and place the remaining five behind another "data name" card with a "5" in columns 29 - 30.

2. Error Messages

At a number of points in the program, checks are made on various input data to determine if it is in the correct form. If there is an error the program will print out a message. Generally, the program will attempt to assign a default value so as to continue with the run. These messages may appear anywhere in the output and usually give an indication of where the problem occurred. These messages are listed below, followed by an explanation of the cause.

EXECUTE CARD MISSING OR OUT OF ORDER

The first card of the execution list must be the EXECUTE card which includes the sensor number. If the first card read is not the EXECUTE card then the above message is printed and the sensor number defaults to the value "1".

INVALID SENSOR NUMBER - (value)

Valid sensor numbers range from "1" to "7". Any sensor number input outside this range will result in the above message. The invalid sensor number read is printed. The sensor number defaults to "1".

INVALID DATA NAME - (name)

The name read does not match one of the legal data names given in Table 1. The invalid name read is printed.

VARIABLE NUMBER OUTSIDE RANGE OF DATA NAME ARRAY - (value)

The variable number in columns 9 - 10 of a "data modification" card is outside the range of the array reserved for the data name appearing on the previous "data name" card. The read value is printed.

MORE MODIFICATIONS THAN LISTED

This refers only to the data modification "listing." It does not imply an error in the run. Space is reserved to print out a maximum of 10 data modifications per "data name" card. If more than 10 "data modification" cards are required for one "data name" card, the user should place the remaining "data modification" cards after another "data name" card containing the same data name as for the first 10 modifications. For many modifications it may be more efficient to repunch that section of the library.

DEPRESSION ANGLE TOO STEEP, PHID RESET = (value)

This message is applicable to all non-radar, fixed sensor depression angle runs. If the input depression angle, PHID, is too large, the computed dive profile point may be reached before the target enters the sensor footprint. The program will then set the angle back to where the target just enters the footprint when the dive profile point is reached. The new angle, in degrees, is printed.

OFFSET (Y) GREATER THAN YMAX. SET Y = (value)

This message is applicable to all models using a fixed sensor depression angle with the exception of the Synthetic Aperture Radar model. For a given depression angle, horizontal beamwidth, and vertical beamwidth, a ground sensor footprint is calculated. The cross-track distance of the footprint at the leading edge is calculated as YMAX. If the input offset (Y) is greater than YMAX, the target never passes into the ground footprint. If such a condition is encountered, Y defaults to the value printed.

ONLY POSITIVE OFFSETS CONSIDERED, USED ABS. VALUE

Offsets are considered to be symmetric with respect to flight path. Therefore, if a negative offset is input the program will convert it to its absolute value.

TUBE SATURATION, NEW FNUM = (value)

Illumination level great enough to saturate tube. Lens is stopped down and another trial is made to see if the illumination is within tube operating limits.

THRESHOLD SPEED IS GREATER THAN 1/2 BLIND SPEED

The MTI filter threshold velocity exceeds one-half the calculated value of blind speed. This means that the filter has no passband and that detection is impossible.

AN MF VALUE WAS NOT INPUT SO MF IS SET - 12

For Forward-Looking Radar (MTI mode) a filter rolloff value, MF, was not input (or input as zero). The program defaults to a value of 12.

Y TOO SMALL, INCREASED TO BE 1.5 SWATH WIDTHS

The offset, Y, is measured to the far edge of the swath width. No information is obtained from the area below or close to the aircraft. For offsets less than 1 - 1/2 swath width, Y is reset to 1.5 times the input swath width. Typical offset values for SAR are on the order of 10 - 30 nautical miles.

3. Standard Model Output

There are basically three types of standard model output generated by a computer run. All three may not be applicable to all sensor models. The discussion below and reference to Figure 4 should familiarize the user with this output. Standard model output follows the list of data modifications (if any).

a. Variable Description

The first lines in this output section (Figure 4, lines 12 - 27) for any run consist of a brief description of the variables which head a given column. Further description may be found in Volume I of this report.

```
         SENSOR NUMBER = 2
 1= VISUAL OBSERVER
 2= FORWARD-LOOKING INFRARED
 3= ACTIVE (ILLUMINATED) TV
 4= PASSIVE (DAYLIGHT) TV
 5= FORWARD-LOOKING RADAR,MTI
 6= FORWARD-LOOKING RADAR,NON-MTI
 7= SYNTHETIC APERTURE RADAR

NAME   IVAR     VALUE
OPER     3     .5000E+04
OPER     4     .2000E+03

I   = MINIMUM LAUNCH POINT
II  = DIVE BEGINS
III= (DUMMY POSITION)
IV  = TARGET PASSES OUT OF FOV(FIXED DEPRESSION ANGLE)
V   = SEARCH ALTITUDE ACHIEVED
VI  = CLIMB TO ALTITUDE BEGINS

X    = ALONG-TRACK GROUND DISTANCE TO TARGET,FT.
XY   = GROUND RANGE TO TARGET,FT.
TIME= TIME BEFORE LAUNCH,SEC.
PLOS= PROBABILITY TARGET IS WITHIN LOS
PFOV= PROBABILITY TARGET IS WITHIN FOV
P2   = SEARCH TERM PROBABILITY
P3D  = DISCRIMINABILITY(DETECTION)
P3R  = DISCRIMINABILITY(RECOGNITION)
PAD  = CUMULATIVE PROBABILITY OF DETECTION
PAR  = CUMULATIVE PROBABILITY OF RECOGNITION
```

I	II	III	IV	V	VI
2762.	9815.	9815.	9815.	59791.	87019.
2762.	9815.	9815.	9815.	59791.	87019.

X	XY	TIME	PLOS	PFOV	P2	P3D	P3R	PAD	PAR
2963.	2963.	.33	1.000	1.000	1.000	1.000	1.000	1.000	1.000
9773.	9773.	11.66	1.000	1.000	1.000	1.000	.017	1.000	1.000
9993.	9993.	11.99	.995	.997	.983	.150	0.000	.146	0.000
10668.	10668.	12.99	.993	.999	.976	.086	0.000	.149	0.000
11342.	11342.	13.99	.991	1.000	.967	.052	0.000	.149	0.000
12016.	12016.	14.99	.989	1.000	.958	.029	0.000	.149	0.000
12691.	12691.	15.98	.986	1.000	.947	.017	0.000	.149	0.000
13365.	13365.	16.98	.983	1.000	.936	.011	0.000	.149	0.000
14040.	14040.	17.98	.980	1.000	.924	.008	0.000	.149	0.000
14714.	14714.	18.98	.977	1.000	.911	.006	0.000	.149	0.000

Figure 4. Sample Computer Output

b. Flight Profile Points

The first two rows of numbers following the variable description (Figure 4, lines 29 - 30) display profile points along the flight path for all computer runs except the Synthetic Aperture Radar*. The first of these rows represent the along-track ground range, in feet, from target to aircraft position for the various profile points listed. The second row represents ground range, in feet, from target to aircraft position for the same profile points. In case of zero offset these two rows are equivalent.

c. Ranges, Time, and Probabilities

Beneath the rows indicating flight profile (or immediately beneath the variable description for the Synthetic Aperture run) is a row of variables (Figure 4, line 31) which head 10 columns of numerical data. These columns of numerical data represent a combination of ranges, time, and probabilities which correspond to the variable description of whichever variable heads a given column. All the data in one particular row is related in-so-far-as it was all generated by one iteration of the program. Each can be thought of as being a function of the range given in the first column.

4. <u>Variations in Standard Model Output</u>

A few special points can be made regarding the output described in the above sections.

a. Non-Radar Cases

For all non-radar cases the following notes apply:

The first row of numerical output from the "ranges, time, and probabilities" output section was computed at a range which corresponded to one "glimpse" before launch point;

The second of these rows of output was computed at a range which corresponds to the first "glimpse" after the dive profile point; and

The third of these rows of output was computed at a range which corresponds to the first "glimpse" before the dive profile point.

---

* Output generated by a Synthetic Aperture Radar run will be given special consideration in a following subsection.

b. Forward-Looking Radar

As discussed in Volume I to this report, all radar submodels are essentially taken from MARSAM* with technology updates. Users familiar with MARSAM will recall that it utilized a fixed depression angle for radar sensors. Thus, for given horizontal and vertical beamwidths a sensor footprint on the ground was clearly defined. Results at one range between the leading and trailing edges of this footprint were output. In the TATAC version a footprint is clearly defined, however, output will be generated for intermittent ranges from the trailing edge up to the leading edge. The flight profile is straight and level throughout and no dive is assumed. Profile points normally printed thus appear as blanks in the dive phase. Refer to Problem B, Section VI for an example of output from the Forward-Looking Radar models.

c. Synthetic Aperture Radar

The Synthetic Aperture Radar is also a version of a MARSAM submodel. Again, there is no dive profile. Furthermore, the Synthetic Aperture Radar processes the doppler phase history of the illuminated ground area over the entire illumination period to generate the display. All returns from the target are integrated into this display. Thus, only one line of output is necessary to present the results. For an example, see Problem C, Section VI.

C. PROGRAM LIBRARY DATA

The program library consists of a group of DATA declaration statements found in subroutine INPUT1. It contains information needed by the program to describe sensors, target, background, etc. The library may contain more information than is needed by the program for a particular run. A complete list of the library data supplied with the program is given in Section V. Tables 2 through 13 following this discussion define the sets of entries allowed in the library. These sets of library data are:

Operational variables;

Target data;

Background data;

Environmental/terrain data;

Search data;

Visual Observer sensor data;

Forward-Looking Infrared sensor data;

Television sensor data (active);

---

* "Multiple Airborne Reconnaissance Sensor Assessment Model", Honeywell Technical Report ASD-TR-68-3, February 1968, Unclassified.

Television sensor data (passive);

Forward-Looking Radar (MTI) sensor data;

Forward-Looking Radar (non-MTI) sensor data; and

Synthetic Aperture Radar sensor data.

Each of these tables lists the name of the type data which it describes, the number of integers in the data (NI), and the number of floating-point numbers (NF). For each integer and floating-point number used to define the entry, the table lists the type of data, the variable number, the FORTRAN symbol which is to represent the variable, the units in which the data is expressed, and a description of the entry.

TABLE 2. OPERATIONAL VARIABLES

NAME: OPERATIONAL VAR: NI - 1; NF - 14

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	-	-	UNUSED
2	FD	Y	FEET	FLIGHT PATH OFFSET
3	FD	HL	FEET	SEARCH ALTITUDE
4	FD	HP	FEET	PENETRATION ALTITUDE
5	FD	SPD	KNOTS	SPEED IN DIVE
6	FD	SPL	KNOTS	SPEED IN LEVEL FLIGHT
7	FD	SPC	KNOTS	SPEED IN CLIMB
8	FD	SPP	KNOTS	SPEED IN PENETRATION
9	FD	SR1	FEET	SLANT RANGE, MINIMUM LAUNCH
10	FD	SR2	FEET	SLANT RANGE, POP-UP ATTAINED
11	FD	ANCLD	DEG	DIVE ANGLE
12	FD	ANGLC	DEG	CLIMB ANGLE
13	FD			
14	FD	-	-	UNUSED
15	FD	-	-	UNUSED

TABLE 3. TARGET DATA

NAME: TARGET; NI - 3; NF - 17

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	NJ	-	TOTAL NUMBER OF TARGET ELEMENTS
2	ID	-	-	UNUSED
3	ID	-	-	UNUSED
4	FD	DTX	FEET	TARGET ALONG-TRACK DIMENSION
5	FD	DTY	FEET	TARGET CROSS-TRACK DIMENSION
6	FD	DTZ	FEET	TARGET HEIGHT
7	FD	RT	-	TARGET REFLECTANCE
8	FD	DELTT	°K	TARGET TEMPERATURE
9	FD	EMMT	-	TARGET EMISSIVITY
10	FD	SIGTX	m^2	MEDIAN TARGET RCS AT 1 GHZ
11	FD	-	-	UNUSED
12	FD	-	-	UNUSED
13	FD	TSPACE	FEET	SPACING BETWEEN TARGETS
14	FD	PSI	DEG	ANGULAR DIRECTION OF TARGET RELATIVE TO FLIGHT PATH
15	FD	VTT	KNOTS	TARGET VELOCITY
16	FD	-	-	UNUSED
17	FD	-	-	UNUSED
18	FD	-	-	UNUSED
19	FD	-	-	UNUSED
20	FD	-	-	UNUSED

TABLE 4. BACKGROUND DATA

NAME: BACKGROUND; NI - 1; NF - 6

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	-	-	UNUSED
2	FD	RB	-	REFLECTANCE OF BACKGROUND
3	FD	DELTB	°K	TEMPERATURE OF BACKGROUND
4	FD	EMMB	-	EMISSIVITY OF BACKGROUND
5	FD	-	-	UNUSED
6	FD	WP	METERS	EXTENT OF PRIMARY BACKGROUND AROUND TARGET
7	FD	-	-	UNUSED

TABLE 5. ENVIRONMENTAL/TERRAIN DATA

NAME: ENVIRONMENTAL OR TERRAIN; NI - 5; NF - 8

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ICLOUD	-	CLOUD COVERAGE (1 = CLEAR, 2 = PARTIAL OVERCAST, 3 = SOLID OVERCAST)
2	ID	ISUNAN	-	SUN ANGLE ABOVE HORIZON (1 = 90°, 2 = 30°, 3 = 11.5°)
3	ID	ICFLOS	-	SWITCH FOR CLOUD FREE LOS (0 = DO NOT CONSIDER, 1 = use tables)
4	ID	ITAT	-	SWITCH FOR ATMOSPHERIC TRANSMITTANCE (0 = USE ANALYTIC APPROXIMATION, 1 = USE EMPIRICAL DATA)
5	ID	IAZIM	-	SUN AZIMUTH, USED IF ITAT = 1 (1 = 90°, 2 = 0°)
6	FD	RTE	-	REFLECTANCE OF TERRAIN
7	FD	ANGLM	DEG.	AVERAGE MASKING ANGLE
8	FD	RATIO	-	CULTURAL MASKING RATIO
9	FD	VG	NM	METEOROLOGICAL VISIBILITY
10	FD	RH	-	RELATIVE HUMIDITY
11	FD	G	-	MEASURE OF SCENE COMPLEXITY
12	FD	VW	M/SEC	APPARENT RMS VELOCITY DUE TO WIND
13	FD	-	-	

TABLE 6. SEARCH DATA

NAME: SEARCH; NI - 1; NF - 9

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	-	-	UNUSED
2	FD	-	-	UNUSED
3	FD	DSX	FEET	SEARCH LENGTH*
4	FD	DSY	FEET	SEARCH WIDTH*
5	FD	WLOC	FEET	LOC WIDTH
6	FD	SIGX	-	STANDARD DEVIATION OF ALONG-TRACK ERROR
7	FD	SIGY	-	STANDARD DEVIATION OF CROSS-TRACK ERROR
8	FD	-	-	UNUSED
9	FD	-	-	UNUSED
10	FD	-	-	UNUSED

* ENTER -1 for these variables if FOV is to be searched. The visual case uses a unique geometry in search along a LOC as described in App A.

TABLE 7. VISUAL SENSOR DATA

NAME: VISOB; NI - 2; NF - 5

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ICMASK	-	SWITCH FOR COCKPIT MASKING (0 = NO, 1 = YES)
2	ID	-	-	UNUSED
3	FD	VISMAX	FEET	MAXIMUM VISUAL SEARCH RANGE
4	FD	-	-	UNUSED
5	FD	-	-	UNUSED
6	FD	XLAMDA	μm	RESPONSIVE WAVE LENGTH
7	FD	-	-	UNUSED

TABLE 8. FORWARD-LOOKING INFRARED

NAME: FLIR; NI - 2; NF - 10

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ISTYPE	-	SWITCH FOR DEPRESSION ANGLE (1 = VARIABLE, 2 = FIXED)
2	ID	-	-	UNUSED
3	FD	ASPECT	-	RATIO OF THE VERTICAL FOV TO HORIZONTAL FOV
4	FD	THETAV	DEG	VERTICAL FOV (NARROW)
5	FD	THETAV	DEG	VERTICAL FOV (WIDE)
6	FD	PHID	DEG	SENSOR DEPRESSION ANGLE
7	FD	ZK1	-	LINEAR CONSTANT TO MRT CURVE
8	FD	ZK2	-	EXPONENTIAL CONSTANT TO MRT CURVE
9	FD	-	-	UNUSED
10	FD	-	-	UNUSED
11	FD	-	-	UNUSED
12	FD	-	-	UNUSED

TABLE 9. ACTIVE TV SENSOR DATA

NAME: TELEVISION; NI - 2; NF - 24

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ISTYPE	-	SWITCH FOR DEPRESSION ANGLE (1=VARIABLE, 2= FIXED)
2	ID	-	-	UNUSED
3	FD	ASPECT	-	RATIO OF THE VERTICAL FOV TO HORIZONTAL FOV
4	FD	THETAV	DEG	VERTICAL FOV (NARROW)
5	FD	THETAV	DEG	VERTICAL FOV (WIDE)
6	FD	PHID	DEG	SENSOR DEPRESSION ANGLE
7	FD	BAND	HZ	BANDWIDTH
8	FD	DIAG	MM	DIAGONAL OF EFFECTIVE PHOTOSURFACE AREA
9	FD	FNUM	-	F - NUMBER (NARROW FOV)
10	FD	FNUM	-	F - NUMBER (WIDE FOV)
11	FD	GAMMAT	-	SLOPE OF SIGNAL VS/ IRRADIANCE CHARACTERISTIC
12	FD	TRANSM	-	LENS SYSTEM TRANSMITTANCE
13	FD	XFC	FC	SENSITIVITY DATA ILLUMINANCE
14	FD	XI	AMPS	SIGNAL AT H_1
15	FD	XIMAX	AMPS	MAXIMUM SIGNAL CAPABILITY
16	FD	XIP	AMPS	PRE-AMP NOISE
17	FD	XLAMDA	μm	RESPONSIVE WAVE LENGTH
18	FD	XNL	-	5% HORIZONTAL FREQUENCY RESPONSE
19	FD	XNR	-	RASTER COUNT
20	FD	POUT	WATTS	EFFECTIVE POWER OF THE ILLUMINATOR
21	FD	GAIN	-	BACKSCATTER GAIN RELATIVE TO ISOTROPIC ILLUMINATOR WAVE-LENGTH
22	FD	GT	-	TUBE GAIN
23	FD	-	-	UNUSED
24	FD	-	-	UNUSED
25	FD	-	-	UNUSED
26	FD	-	-	UNUSED

TABLE 10. PASSIVE TV

NAME: TELEVISION; NI - 2; NF - 24

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ISTYPE	-	SWITCH FOR DEPRESSION ANGLE (1=VARIABLE, 2= FIXED)
2	ID	-	-	UNUSED
3	FD	ASPECT	-	RATIO OF THE VERTICAL FOV TO HORIZONTAL FOV
4	FD	THETAV	DEG	VERTICAL FOV (NARROW)
5	FD	THETAV	DEG	VERTICAL FOV (WIDE)
6	FD	PHID	DEG	SENSOR DEPRESSION ANGLE
7	FD	BAND	HZ	BANDWIDTH
8	FD	DIAG	MM	DIAGONAL OF EFFECTIVE PHOTOSURFACE AREA
9	FD	FNUM	-	F - NUMBER (NARROW FOV)
10	FD	FNUM	-	F - NUMBER (WIDE FOV)
11	FD	GAMMAT	-	SLOPE OF SIGNAL VS/ IRRADIANCE CHARACTERISTIC
12	FD	TRANSM	-	LENS SYSTEM TRANSMITTANCE
13	FD	XFC	FC	SENSITIVITY DATA ILLUMINANCE
14	FD	XI	AMPS	SIGNAL AT H_i
15	FD	XIMAX	AMPS	MAXIMUM SIGNAL CAPABILITY
16	FD	XIP	AMPS	PRE-AMP NOISE
17	FD	XLAMDA	μm	RESPONSIVE WAVE LENGTH
18	FD	XNL	-	5% HORIZONTAL FREQUENCY RESPONSE
19	FD	XNR	-	RASTER COUNT
20	FD	POUT	WATTS	EFFECTIVE POWER OF THE ILLUMINATOR
21	FD	GAIN	-	BACKSCATTER GAIN RELATIVE TO ISOTROPIC ILLUMINATOR WAVE-LENGTH
22	FD	GT	-	TUBE GAIN
23	FD	-	-	UNUSED
24	FD	-	-	UNUSED.
25	FD	-	-	UNUSED
26	FD	-	-	UNUSED

TABLE 11. FORWARD-LOOKING RADAR, MTI

NAME: FLR; NI - 2; NF - 23

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ISEAS	-	SEASON (1 = WINTER, 2 = SPRING, 3 = SUMMER, 4 = FALL)
2	ID	-	-	UNUSED
3	FD	FMF	db/oct	FILTER ROLLOFF VALUE
4	FD	GO	-	MAXIMUM ANTENNA GAIN
5	FD	PNR	WATTS	RECEIVER NOISE POWER
6	FD	PRF	PPS	PULSE REPETITION RATE
7	FD	PX	WATTS	PEAK TRANSMITTER POWER
8	FD	FLAMDA	METERS	WAVELENGTH
9	FD	WS	DEG/SEC	ANTENNA SCAN RATE
10	FD	PHIM	DEG	ANTENNA DEPRESSION ANGLE TO TOP OF BEAM
11	FD	THETAH	DEG	HORIZONTAL ANTENNA BEAMWIDTH
12	FD	THETAS	DEG	TOTAL HORIZONTAL SCAN ANGLE, MEASURE IN HORIZONTAL PLANE
13	FD	THETAV	DEG	VERTICAL ANTENNA BEAMWIDTH
14	FD	TAUPW	SEC	PULSEWIDTH
15	FD	TAUP1	-	TRANSMITTER PLUMBING ATTENUATION FACTOR
16	FD	TAUP2	-	RECEIVER PLUMBING ATTENUATION FACTOR
17	FD	VF	CPS	DOPPLER SPREAD DUE TO CARRIER FREQUENCY DEVIATION
18	FD	VPRF	CPS	DOPPLER SPREAD DUE TO PRF DEVIATION
19	FD	-	-	UNUSED
20	FD	-	-	UNUSED
21	FD	CAS	-	FRACTION OF TIME ATTENUATION NOT EXCEEDED
22	FD	-	-	UNUSED
23	FD	TSM	db	SIGNAL TO NOISE THRESHOLD
24	FD	-	-	UNUSED
25	FD	VTH	M/SEC	THRESHOLD VELOCITY (RADIAL)

TABLE 12. FORWARD-LOOKING RADAR, NON-MTI

NAME: FLR; NI - 2; NF - 23

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ISEAS	-	SEASON (1 - WINTER, 2 - SPRING, 3 - SUMMER, 4 - FALL)
2	ID	-	-	UNUSED
3	FD	FMF	db/oct	FILTER ROLLOFF VALUE
4	FD	GO	-	MAXIMUM ANTENNA GAIN
5	FD	PNR	WATTS	RECEIVER NOISE POWER
6	FD	PRF	PPS	PULSE REPETITION RATE
7	FD	PX	WATTS	PEAK TRANSMITTER POWER
8	FD	FLAMDA	METERS	WAVELENGTH
9	FD	WS	DEG/SEC	ANTENNA SCAN RATE
10	FD	PHIM	DEG	ANTENNA DEPRESSION ANGLE TO TOP OF BEAM
11	FD	THETAH	DEG	HORIZONTAL ANTENNA BEAMWIDTH
12	FD	THETAS	DEG	TOTAL HORIZONTAL SCAN ANGLE, MEASURE IN HORIZONTAL PLANE
13	FD	THETAV	DEG	VERTICAL ANTENNA BEAMWIDTH
14	FD	TAUPW	SEC	PULSEWIDTH
15	FD	TAUP1	-	TRANSMITTER PLUMBING ATTENUATION FACTOR
16	FD	TAUP2	-	RECEIVER PLUMBING ATTENUATION FACTOR
17	FD	VF	CPS	DOPPLER SPREAD DUE TO CARRIER FREQUENCY DEVIATION
18	FD	VPRF	CPS	DOPPLER SPREAD DUE TO PRF DEVIATION
19	FD	-	-	UNUSED
20	FD	-	-	UNUSED
21	FD	CAS	-	FRACTION OF TIME ATTENUATION NOT EXCEEDED
22	FD	-	-	UNUSED
23	FD	TSM	db	SIGNAL TO NOISE THRESHOLD
24	FD	-	-	UNUSED
25	FD	VTH	M/SEC	THRESHOLD VELOCITY (RADIAL)

TABLE 13. SYNTHETIC APERTURE RADAR

NAME: SAR; NI - 2; NF - 23

VARIABLE NUMBER	VARIABLE TYPE	FORTRAN SYMBOL	UNITS	DESCRIPTION
1	ID	ISEAS	-	SEASON (1=WINTER, 2=SPRING, 3=SUMMER, 4=FALL)
2	ID	-	-	UNUSED
3	FD	DX	METERS	AZIMUTH GROUND RESOLUTION
4	FD	GZER	-	UNUSED
5	FD	PNR	WATTS	RECEIVER NOISE POWER
6	FD	PRF	PPS	PULSE REPETITION RATE
7	FD	PX	WATTS	PEAK TRANSMITTER POWER
8	FD	RC	-	CHIRP RATIO
9	FD	SNT	db	SIGNAL TO NOISE THRESHOLD
10	FD	-	-	UNUSED
11	FD	FLAMDA	METERS	WAVELENGTH
12	FD	-	-	UNUSED
13	FD	-	-	UNUSED
14	FD	THETAV	DEG	ANTENNA VERTICAL FOV
15	FD	TAUPW	SEC	PULSEWIDTH
16	FD	TAUP1	-	TRANSMITTER PLUMBING ATTENUATION FACTOR
17	FD	TAUP2	-	RECEIVER PLUMBING ATTENUATION FACTOR
18	FD	-	-	UNUSED
19	FD	ASPECT	-	
20	FD	SQ	DEG	SQUINT ANGLE
21	FD	CAS	-	FRACTION OF TIME ATTENUATION NOT EXCEEDED
22	FD	SL1	-	FIRST SIDELOBE LEVEL WITH RESPECT TO MAINLOBE
23	FD	SL2	-	SECOND SIDELOBE LEVEL WITH RESPECT TO MAINLOBE
24	FD	SL3	-	THIRD SIDELOBE LEVEL WITH RESPECT TO MAINLOBE
25	FD	WS	NM	GROUND SWATH WHICH CAN BE PROCESSED IN REAL TIME

D. FIXED DATA

The fixed data consists of a group of DATA declaration statements found in subroutine INPUT1. It contains data which is constant for all sensor models in the program. It should not be modified by the user unless the program itself is modified or the user wishes to change some of the contained data which is discussed below. The only method of modifying the fixed data is to repunch the cards containing the data. As a reference for the discussion of this subsection, see the Fixed Data List in Section IV.

The first portion of the fixed data contains 50 integers which are described in Table 14. This portion of the fixed data is addressed as array IC in the program.

TABLE 14. INTEGERS IN FIXED DATA ARRAY IC

INTEGER NUMBER	DESCRIPTION	TYPE OF DATA*
1	MEMORY ALLOCATED FOR OPERATIONAL VARIABLES	ID
2	MEMORY ALLOCATED FOR OPERATIONAL VARIABLES	FD
3	MEMORY ALLOCATED FOR A TARGET ENTRY	ID
4	MEMORY ALLOCATED FOR A TARGET ENTRY	FD
5	MEMORY ALLOCATED FOR A BACKGROUND ENTRY	ID
6	MEMORY ALLOCATED FOR A BACKGROUND ENTRY	FD
7	MEMORY ALLOCATED FOR AN ENVIRONMENTAL/TERRAIN ENTRY	ID
8	MEMORY ALLOCATED FOR AN ENVIRONMENTAL/TERRAIN ENTRY	FD
9	MEMORY ALLOCATED FOR SEARCH VARIABLES	ID
10	MEMORY ALLOCATED FOR SEARCH VARIABLES	FD
11	MEMORY ALLOCATED FOR A VISUAL SENSOR ENTRY	ID
12	MEMORY ALLOCATED FOR A VISUAL SENSOR ENTRY	FD
13	MEMORY ALLOCATED FOR A FLIR SENSOR ENTRY	ID
14	MEMORY ALLOCATED FOR A FLIR SENSOR ENTRY	FD
15	MEMORY ALLOCATED FOR AN ACTIVE TV SENSOR ENTRY	ID
16	MEMORY ALLOCATED FOR AN ACTIVE TV SENSOR ENTRY	FD
17	MEMORY ALLOCATED FOR A PASSIVE TV SENSOR ENTRY	ID
18	MEMORY ALLOCATED FOR A PASSIVE TV SENSOR ENTRY	FD
19	MEMORY ALLOCATED FOR A MTI FLR SENSOR ENTRY	ID
20	MEMORY ALLOCATED FOR A MTI FLR SENSOR ENTRY	FD
21	MEMORY ALLOCATED FOR A NON-MTI FLR SENSOR ENTRY	ID
22	MEMORY ALLOCATED FOR A NON-MTI FLR SENSOR ENTRY	FD
23	MEMORY ALLOCATED FOR A SAR SENSOR ENTRY	ID
24	MEMORY ALLOCATED FOR A SAR SENSOR ENTRY	FD
25-50	UNUSED LOCATIONS	-

* ID - INTEGER; FD - FLOATING-POINT

The second portion of the fixed data contains 50 additional integers which are described in Table 15. This portion of the fixed data is addressed as array ICT in the program.

TABLE 15. INTEGERS IN FIXED DATA ARRAY ICT

INTEGER NUMBER	DESCRIPTION *
1	MEMORY ALLOCATED FOR TABLE NUMBER 1 OF FDTAB ARRAY
2	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 1
3	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 1
4	MEMORY ALLOCATED FOR TABLE NUMBER 2 OF FDTAB ARRAY
5	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 2
6	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 2
7	MEMORY ALLOCATED FOR TABLE NUMBER 3 OF FDTAB ARRAY
8	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 3
9	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 3
10	MEMORY ALLOCATED FOR TABLE NUMBER 4 OF FDTAB ARRAY
11	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 4
12	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 4
13	MEMORY ALLOCATED FOR TABLE NUMBER 5 OF FDTAB ARRAY
14	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 5
15	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 5
16	MEMORY ALLOCATED FOR TABLE NUMBER 6 OF FDTAB ARRAY
17	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 6
18	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 6
19	MEMORY ALLOCATED FOR TABLE NUMBER 7 OF FDTAB ARRAY
20	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 7
21	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 7
22	MEMORY ALLOCATED FOR TABLE NUMBER 8 OF FDTAB ARRAY
23	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 8
24	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 8
25	MEMORY ALLOCATED FOR TABLE NUMBER 9 OF FDTAB ARRAY
26	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 9
27	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 9
28	MEMORY ALLOCATED FOR TABLE NUMBER 10 OF FDTAB ARRAY
29	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 10
30	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 10

TABLE 15. INTEGERS IN FIXED DATA ARRAY ICT (Continued)

INTEGER NUMBER	DESCRIPTION *
31	MEMORY ALLOCATED FOR TABLE NUMBER 11 OF FDTAB ARRAY
32	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 11
33	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 11
34	MEMORY ALLOCATED FOR TABLE NUMBER 12 OF FDTAB ARRAY
35	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 12
36	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 12
37	MEMORY ALLOCATED FOR TABLE NUMBER 13 OF FDTAB ARRAY
38	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 13
39	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 13
40	MEMORY ALLOCATED FOR TABLE NUMBER 14 OF FDTAB ARRAY
41	NUMBER OF FIRST INDEPENDENT VARIABLES IN TABLE 14
42	NUMBER OF SECOND INDEPENDENT VARIABLES IN TABLE 14
43-50	UNUSED LOCATIONS

* A LIST OF THE TABLE NUMBERS AND DESCRIPTION FOR THE FDTAB ARRAY CAN BE FOUND IN TABLE 16.

The third portion of the fixed data contains all the tables used in the program. The original 14 tables are described in Table 16. This portion of the fixed data is addressed by the array FDTAB.

The floating-point, third portion of the fixed data contains all the tables used in the program. The original 14 tables are described in Table 16. This portion of the fixed data is addressed by the array FDTAB.

TABLE 16. TABLES IN FIXED DATA ARRAY FDTAB

TABLE NUMBER	TABLE SIZE	DESCRIPTION
1	88	TABULAR VALUES FOR E_2 (Z) where $E_2(Z) = \int_1^{\infty} e^{-zt}/t^2 \, dt$
2	8	PROBABILITY OF CLOUD-FREE LOS VERSUS ZENITH ANGLE (DEG.) FOR ALTITUDES LESS THAN OR EQUAL TO 5000 FT.
3	8	PROBABILITY OF CLOUD-FREE LOS VERSUS ZENITH ANGLE (DEG.) FOR ALTITUDES GREATER THAN 5000 FEET
4	15	BACKGROUND RCS (M^{-2}) VERSUS DEPRESSION ANGLE (DEG.) AND RADAR BAND.
5	31	ATTENUATION RATE VERSUS RAIN RATE (MM/HR) AND RADAR BAND.
6	14	ALTITUDE OF CLOUD LAYER (CEILING OR FLOOR) VERSUS SEASON
7	49	LIQUID WATER CONTENT (g/m^3) VERSUS SEASON AND OCTALS OF CLOUD COVER
8	49	PERCENTAGE FREQUENCY OF CLOUD COVER VERSUS SEASON AND OCTALS OF CLOUD COVER.
9	39	FREQUENCY OF RAINFALL RATE VERSUS SEASON AND RAINFALL RATE (MM/HR.) SLANT RANGE FROM BOTTOM OF CLOUD COVER TO GROUND GREATER THAN OR EQUAL TO 10 KM.
10	39	FREQUENCY OF RAINFALL RATE VERSUS SEASON AND RAINFALL RATE (MM/HR) SLANT RANGE FROM BOTTOM OF CLOUD COVER TO GROUND LESS THAN 10 KM.
11	181	DIRECTIONAL PATH REFLECTANCE VERSUS ALTITUDE (FT.) AND SLANT RANGE (FT.) FOR 90° AZIMUTH.
12	181	DIRECTIONAL PATH REFLECTANCE VERSUS ALTITUDE (FT) AND SLANT RANGE (FT) FOR 0° AZIMUTH.
13	8	REFLECTANCE FACTOR VERSUS ZENITH ANGLE (DEG.) FOR 90° AZIMUTH.
14	8	REFLECTANCE FACTOR VERSUS ZENITH ANGLE (DEG.) FOR 0° AZIMUTH.

III. LOGICAL STRUCTURE

This program is arranged modularly with respect to the sensor models. The control section is independent, allowing sensor models to be added or changed without changing the whole program.

Four main data arrays form the basic structure of the TATAC model. They are the IC, ICT, FDTAB, and FD arrays. The first three array names stored the fixed data as discussed in the previous section. These three arrays (IC, ICT, and FDTAB) are contained in the labeled common blocks ARRSET, TABSET, and TABLES, respectively. The information is available to all programs in which the common block appears.

The array FD contains the input library data used by the program to execute the model for a particular run. This array is contained in the common block ARRAYS. The information is passed through the program using the subroutine call statements. Integer values in this array are addressed by the array ID which shares storage with the FD array through the use of an EQUIVALENCE statement. In the main program, the data in the FD array is identified by pointers to the section reserved for each data type. These pointers are listed in Table 17.

TABLE 17. DATA IDENTIFICATION POINTERS

FORTRAN SYMBOL	TYPE OF DATA *	TYPE OF DATA IDENTIFIED
IIO	ID	OPERATIONAL VARIABLES
IFO	FD	OPERATIONAL VARIABLES
IIT	ID	TARGET
IFT	FD	TARGET
IIB	ID	BACKGROUND
IFB	FD	BACKGROUND
IIEVT	ID	ENVIRONMENTAL/TERRAIN
IFEVT	FD	ENVIRONMENTAL/TERRAIN
IITS	ID	SEARCH
IFTS	FD	SEARCH
ISETID	ID	SENSORS
ISETFD	FD	SENSORS

The data is passed through the call handler using the following procedure:

```
CALL TEST (......ID(IIO),FD(IFO),FD(IFT),....)
```

and is received in the subroutine as follows:

```
SUBROUTINE TEST (.....IDO, FDO,FDT,......)

DIMENSION IDO(1),FDO(1),FDT(1)
```

Table 18 lists the internal representatives for all the sections of the data used in the subroutines.

TABLE 18. INTERNAL ARRAY NAMES

ARRAY NAME	TYPE OF DATA*	LIBRARY DATA
IDO	ID	OPERATIONAL VARIABLES
FDO	FD	OPERATIONAL VARIABLES
IDT	ID	TARGET
FDT	FD	TARGET
IDB	ID	BACKGROUND
FDB	FD	BACKGROUND
IDEVT	ID	ENVIRONMENTAL/TERRAIN
FDEVT	FD	ENVIRONMENTAL/TERRAIN
IDTS	ID	SEARCH
FDTS	FD	SEARCH
IDS	ID	SENSORS
FDS	FD	SENSORS

* ID - INTEGER: FD - FLOATING-POINT NUMBER.

IV. FIXED DATA LISTING

The following is a listing of the TATAC fixed data. For an explanation of listing content, see Section II-D of this Volume.

IC ARRAY

```
  1   2  16  19  36  37  43  48  56  57
 66  68  73  75  85  87 111 113 136 138
161 163 186 188 211 211 211 211 211 211
211 211 211 211 211 211 211 211 211 211
211 211 211 211 211 211 211 211 211 211
```

ICT ARRAY

```
  1  44   0  89   4   0  97   4   0 105
  3   3 120   3   7 151   2   4 165   4
  9 214   4   9 263   4   7 302   4   7
341   6  25 522   6  25 703   4   0 711
  4   0 719 719 719 719 719 719 719 719
```

FDTAB ARRAY

```
0.00  .05  .10  .15  .20  .25  .30  .35  .40  .45
 .50  .55  .60  .65  .70  .75  .80  .85  .90  .95
1.00 1.05 1.10 1.15 1.20 1.25 1.30 1.35 1.40 1.45
1.50 1.55 1.60 1.65 1.70 1.75 1.80 1.85 1.90 1.95
2.00 2.50 4.00 5.00
1.0000000   .8278345   .7225450   .6410387   .5742006
 .5177301   .4691153   .4267127   .3893680   .3562290
 .3266439   .3000996   .2761839   .2545597   .2349471
 .2171109   .2008517   .1859986   .1724041   .1599404
 .1484955   .1379713   .1282811   .1193481   .1111041
 .1034881   .0964455   .0899275   .0838899   .0782930
 .0731008   .0682807   .0638032   .0596413   .0557706
 .0521687   .0488153   .0456915   .0427803   .0400660
 .0375343   .0197976   .0031982   .0009964
100. 120. 140. 180.  .3000  .4050  .4675  .5000
100. 120. 150. 180.  .5130  .6330  .6930  .7000
  5.  40.  75.   1.   2.   3. -43. -29. -25. -39.
-25. -20. -23. -17. -10.
  .0320000   .0180000   .0086000  0.0000000   .2500000
  .5100000  2.5400000  6.3500000 12.7000000 25.4000000
 0.0000000   .0000014   .0000042   .0000270   .0000920
  .0002200   .0005600  0.0000000   .0000086   .0000200
  .0001400   .0003500   .0009000   .0019000  0.0000000
  .0000620   .0001100   .0008500   .0016000   .0043000
  .0076000
  2.    1.    1.    2.    3.    4. 1823. 3414. 4816. 4333.
 366.  914. 1829. 1524.
1.000 2.000 3.000 4.000 0.000  .125  .250  .375  .500  .625
 .750  .875 1.000 0.000  .250  .250  .250  .400  .500  .500
 .500  .500 0.000  .200  .200  .200  .400  .400  .600  .600
 .600 0.000  .200  .200  .400  .400  .600  .600  .800 1.000
 .200  .200  .200  .400  .400  .600  .600  .600
1.000 2.000 3.000 4.000 0.000  .125  .250  .375  .500  .625
 .750  .875 1.000  .141  .061  .039  .038  .028  .038  .053
 .140  .462  .226  .120  .057  .078  .050  .067  .091  .105
 .207  .154  .177  .087  .092  .071  .085  .095  .149  .090
 .216  .113  .052  .052  .046  .048  .062  .125  .286
  .2160    .1130    .0520    .0520    .0460    .0480    .0620    .1250
 .2860
```

IC ARRAY (Continued)

```
1.0000  2.0000  3.0000  4.0000  0.0000   .2500   .5100  2.5400
6.3500 12.7000 25.4000   .9240   .0560   .0200  0.0000  0.0000
0.0000  0.0000   .9290   .0480   .0210   .0020  0.0000  0.0000
0.0000   .8937   .0580   .0390   .0080   .0010   .0003  0.0000
 .8890   .0640   .0420   .0050  0.0000  0.0000  0.0000
1.00000   2.00000   3.00000   4.00000   0.00000    .25000    .51000
2.54000   6.35000  12.70000  25.40000    .98110    .00780    .00640
 .00300    .00140    .00030   0.00000    .97940    .00790    .00680
 .00370    .00180    .00040   0.00000    .97710    .00680    .00660
 .00550    .00320    .00080   0.00000    .97762    .00690    .00680
 .00510    .00290    .00068   0.00000
1000.00   1500.00   5000.00  10000.00  20000.00  30000.00      1.00
   1.50      2.00      3.00      4.00      5.00      6.00      7.00
   8.00      9.00     10.00     15.00     20.00     25.00     30.00
  35.00     40.00     50.00     60.00     70.00     80.00     90.00
 100.00    150.00    200.00       .02       .02       .03       .06
    .08       .12       .16       .20       .24       .30       .35
    .73      1.35      2.25      3.50      5.40      8.40     21.50
  90.00     90.00     90.00     90.00     90.00     90.00     90.00
   0.00       .02       .03       .06       .09       .13       .17
    .22       .28       .34       .42       .88      1.65      3.00
   5.60     10.50     23.00     80.00     80.00     80.00     80.00
  80.00     80.00     80.00     80.00      0.00      0.00      0.00
   0.00      0.00       .08       .10       .13       .16
   .20    .24    .51    .94   1.55   2.50   3.80   5.80  13.00
 38.50 100.00 100.00 100.00 100.00 100.00 100.00   0.00   0.00
  0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00    .09
   .18    .31    .49    .72   1.00   1.36   2.35   3.85   6.30
 10.00  16.00  25.50 100.00 100.00   0.00   0.00   0.00   0.00
  0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00    .14
   .20    .27    .37    .50    .84   1.30   1.92   2.75   3.75
  5.20  21.00 100.00   0.00   0.00   0.00   0.00   0.00   0.00
  0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00    .22
   .27    .34    .56    .90   1.38   2.05   2.88   3.95  12.00
 26.00
1000.00   1500.00   5000.00  10000.00  20000.00  30000.00      1.00
   1.50      2.00      3.00      4.00      5.00      6.00      7.00
   8.00      9.00     10.00     15.00     20.00     25.00     30.00
  35.00     40.00     50.00     60.00     70.00     80.00     90.00
 100.00    150.10    200.00       .01       .03       .04       .09
    .16       .24       .33       .47       .53      8.65       .78
   1.70      3.10      U.20      8.20     12.25     17.75     38.00
  84.00    100.00    100.00    100.00    100.00    100.00    100.00
   0.00       .02       .04       .08       .15       .22       .32
    .44       .56       .70       .88      2.07      4.10      7.40
  12.50     20.50     32.00     72.00    100.00    100.00    100.00
 100.00    100.00    100.00    100.00      0.00      0.00      0.00
   0.00      0.00       .07       .11       .15       .19
   .25    .32    .80   1.60   2.85   4.80   7.60  12.25  30.00
 66.00 100.00 100.00 100.00 100.00 100.00 100.00   0.00   0.00
  0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00    .10
   .21    .40    .67   1.05   1.55   2.23   4.10   7.00  11.20
 18.20  31.50  60.00 120.00 120.00   0.00   0.00   0.00   0.00
  0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00    .14
   .22    .34    .46    .64   1.10   1.75   2.60   3.90   5.40
  7.60  34.00 160.00   0.00   0.00   0.00   0.00   0.00   0.00
  0.00   0.00   0.00   0.00   0.00   0.00   0.00   0.00    .22
   .28    .36    .58    .89   1.30   1.90   2.70   3.65  15.00
 47.00
 95. 120. 150. 180. 1.95 1.10  .65 1.00
 95. 120. 150. 180. 2.78 1.78 1.10 1.00
 0.  0.  0.  0.  0.  0.  0.
```

V. LIBRARY DATA LISTING

The following is a listing of the TATAC library data. For an explanation of listing content, see Section II-C of this Volume.

FD ARRAY

```
 0
0.            .4000E+04    .5000E+03    .4000E+03    .4000E+03
 .4000E+03    .4000E+03    .3100E+04    .6000E+05    .2700E+02
 .1000E+02    .3330E+00   0.           0.

 10    0    0
 .2050E+02    .1070E+02    .8800E+01    .1600E+00    .3060E+03
 .9000E+00    .1170E+02    .1170E+02    .1170E+02    .9000E+02
0.            .2500E+02   0.           0.           0.
0.           0.

 0
0.8000E-01    .3000E+03    .9100E+00   0.            .6000E+01
0.

 2    2    0    0    0
 .1000E+00    .5000E+01   0.            .6000E+01    .4000E+00
 .4000E+01    .2500E+00   0.

 0
0.            .3600E+05    .1000E+03    .2000E+02    .2550E+03
 .2550E+03   0.           0.           0.

 1    0
 .3600E+05   0.           0.            .5500E+00   0.

 1    0
 .1333E+01    .1860E+01    .7440E+01    .3000E+02    .3060E-01
 .1039E-01   0.           0.           0.           0.

 1    0
 .1000E+01    .1000E+01    .1000E+01    .3000E+02    .4500E+07
 .2500E+02    .5600E+01    .5600E+01    .1000E+01    .6000E+00
 .4500E-02    .2000E-04    .2000E-01    .2000E-08    .8400E+00
 .4000E+03    .5110E+03    .5000E+02    .2400E+00    .1000E+01
0.           0.           0.           0.

 1    0
 .1000E+01    .1000E+01    .4000E+01    .3000E+02    .9250E+07
 .1570E+02    .5600E+01    .5600E+01    .1000E+01    .6000E+00
 .4500E-02    .8000E-08    .8000E-05    .7000E-08    .5500E+00
 .6000E+03    .5110E+03   0.           0.            .1000E+01
0.           0.           0.
```

FD ARRAY (Continued)

```
   1    0
 .2400E+02  .1000E+04  .2000E-12  .3600E+04  .6000E+05
 .3000E-01  .9000E+02  .1000E+02  .2500E+01  .9000E+02
 .3000E+02  .2000E-05  .5000E+00  .8000E+00 0.
0.         0.         0.          .9500E+00 0.
 .1500E+02 0.          .8000E+01

   1    0
 .2400E+02  .1000E+04  .2000E-12  .3600E+04  .6000E+05
 .3000E-01  .9000E+02  .1000E+02  .2500E+01  .9000E+02
 .3000E+02  .2000E-05  .5000E+00  .8000E+00 0.
0.         0.         0.          .9500E+00 0.
 .1500E+02 0.          .8000E+01

   1    0
 .6000E+01  .1000E+04  .2000E-12  .3600E+04  .6000E+05
 .5000E+02  .1500E+02 0.          .3000E-01 0.
0.          .3000E+02  .1000E-05  .5000E+00  .8000E+00
0.          .1000E+01  .1000E+02  .9500E+00  .3600E+02
 .3600E+02  .3600E+02  .1000E+01
```

VI. SAMPLE PROBLEMS

This section, containing sample problems, is prcvided to assist the user in setting up problems for interpreting the outputs of the computer model. It does not present a comprehensive list of all possible representative uses of the model. Each sample briefly describes the problem, presents the execution list and computer printout, and discusses the results.

A. VISUAL OBSERVER (VISOB) - SAMPLE PROBLEM A

This sample problem exemplifies the use of the Visual Observer model. The aircraft search altitude is 6,000 feet with a meteorological visibility of 4 nautical miles.

The specific variables which were entered into the program are shown in the execution list of Figure 5. Two "data name" cards were required with one "data modification" card following each. All other stored library values remain unchanged.

The results of the computer run can be seen from the computer output of Figure 6. The cumulative probability of detection (PAD) was .196 up to the dive point and 1.0 at the minimum launch point.

```
EXECUTE
OPERATIONAL VAR               1
        3 .6000E+04
ENVIRONMENTAL                 1
        9 .4000E+01
ENDRUN
```

Figure 5. Execution List - Sample Problem A

```
            SENSOR NUMBER = 1
 1= VISUAL OBSERVER
 2= FORWARD-LOOKING INFRARED
 3= ACTIVE (ILLUMINATED) TV
 4= PASSIVE (DAYLIGHT) TV
 5= FØRWARD-LOOKING RADAR,MTI
 6= FØRWARD-LOOKING RADAR,NON-MTI
 7= SYNTHETIC APERTURE RADAR

NAME   IVAR    VALUE
ØPER     3    .6000E+04
ENVI     9    .4000E+01

I   = MINIMUM LAUNCH POINT
II  = DIVE BEGINS
III= (DUMMY POSITION)
IV  = TARGET PASSES OUT OF FOV(FIXED DEPRESSION ANGLE)
V   = SEARCH ALTITUDE ACHIEVED
VI  = CLIMB TØ ALTITUDE BEGINS

X    = ALØNG-TRACK GROUND DISTANCE TO TARGET,FT.
XY   = GROUND RANGE TO TARGET,FT.
TIME= TIME BEFORE LAUNCH,SEC.
PLØS= PROBABILITY TARGET IS WITHIN LØS
PFØV= PROBABILITY TARGET IS WITHIN FØV
P2   = SEARCH TERM PROBABILITY
P3D  = DISCRIMINABILITY(DETECTION)
P3R  = DISCRIMINABILITY(RECOGNITION)
PAD  = CUMULATIVE PROBABILITY OF DETECTION
PAR  = CUMULATIVE PROBABILITY OF RECOGNITION
```

	I	II	III	IV	V	VI
	2762.	11778.	11778.	26166.	36000.	36000.
	2762.	11778.	11778.	26166.	36000.	36000.

X	XY	TIME	PLOS	PFOV	P2	P3D	P3R	PAD	PAR
2963.	2963.	.33	1.000	1.000	1.000	1.000	1.000	1.000	1.000
11777.	11777.	14.99	1.000	1.000	1.000	.217	.005	1.000	1.000
12001.	12001.	15.32	1.000	1.000	1.000	.196	.002	.196	.002
12676.	12676.	16.32	1.000	1.000	1.000	.146	0.000	.196	.002
13350.	13350.	17.32	1.000	1.000	1.000	.109	0.000	.196	.002
14024.	14024.	18.32	1.000	1.000	1.000	.083	0.000	.196	.002
14699.	14699.	19.31	1.000	1.000	1.000	.064	0.000	.196	.002
15373.	15373.	20.31	1.000	1.000	1.000	.050	0.000	.196	.002
16048.	16048.	21.31	1.000	1.000	1.000	.040	0.000	.196	.002
16722.	16722.	22.31	1.000	1.000	1.000	.032	0.000	.196	.002
17397.	17397.	23.31	1.000	1.000	1.000	.027	0.000	.196	.002
18071.	18071.	24.31	1.000	1.000	1.000	.023	0.000	.196	.002
18746.	18746.	25.31	1.000	1.000	1.000	.019	0.000	.196	.002
19420.	19420.	26.31	1.000	1.000	1.000	.017	0.000	.196	.002
20094.	20094.	27.31	1.000	1.000	1.000	.015	0.000	.196	.002
20769.	20769.	28.30	1.000	1.000	1.000	.013	0.000	.196	.002
21443.	21443.	29.30	1.000	1.000	1.000	.012	0.000	.196	.002
22118.	22118.	30.30	1.000	1.000	1.000	.011	0.000	.196	.002
22792.	22792.	31.30	1.000	1.000	1.000	.010	0.000	.196	.002
23467.	23467.	32.30	1.000	1.000	1.000	.009	0.000	.196	.002
24141.	24141.	33.30	1.000	1.000	1.000	.008	0.000	.196	.002
24815.	24815.	34.30	1.000	1.000	1.000	.008	0.000	.196	.002
25490.	25490.	35.30	1.000	1.000	1.000	.007	0.000	.196	.002
26164.	26164.	36.30	1.000	1.000	1.000	.007	0.000	.196	.002
26839.	26839.	37.30	.920	1.000	1.000	.007	0.000	.196	.002
27513.	27513.	38.29	.915	1.000	1.000	.007	0.000	.196	.002

Figure 6. Computer Output - Sample Problem A

B. FORWARD-LOOKING RADAR (FLR) - SAMPLE PROBLEM B

This sample problem exemplifies the use of the Forward-Looking Radar model with MTI mode. The radar is operated against a target which has a median RCS of 50 M^2.

The specific variables entered into the program are shown in the execution list of Figure 7. Only one "data name" card and one "data modification" card were required. All other stored library values remain unchanged.

The results of the computer run can be seen from the computer output of Figure 8. As can be seen by the flight profile printout (lines 28-29), the aircraft achieved search altitude at a ground range of 22,689 feet and the target passed out of the sensor field of view at a ground range of 4768 feet. The cumulative probability of detection (PAD) reached a maximum of .994 which was achieved on the first "glimpse" at 22,528 feet. Closing to the target did not improve the PAD and at a ground range below 8365 feet the MTI could not discriminate the target for detection. This situation occurred due to the following reasons:

1. The area of the resolution cell increases as the range decreases, and

2. The unit background RCS increases with increasing depression angle to target.

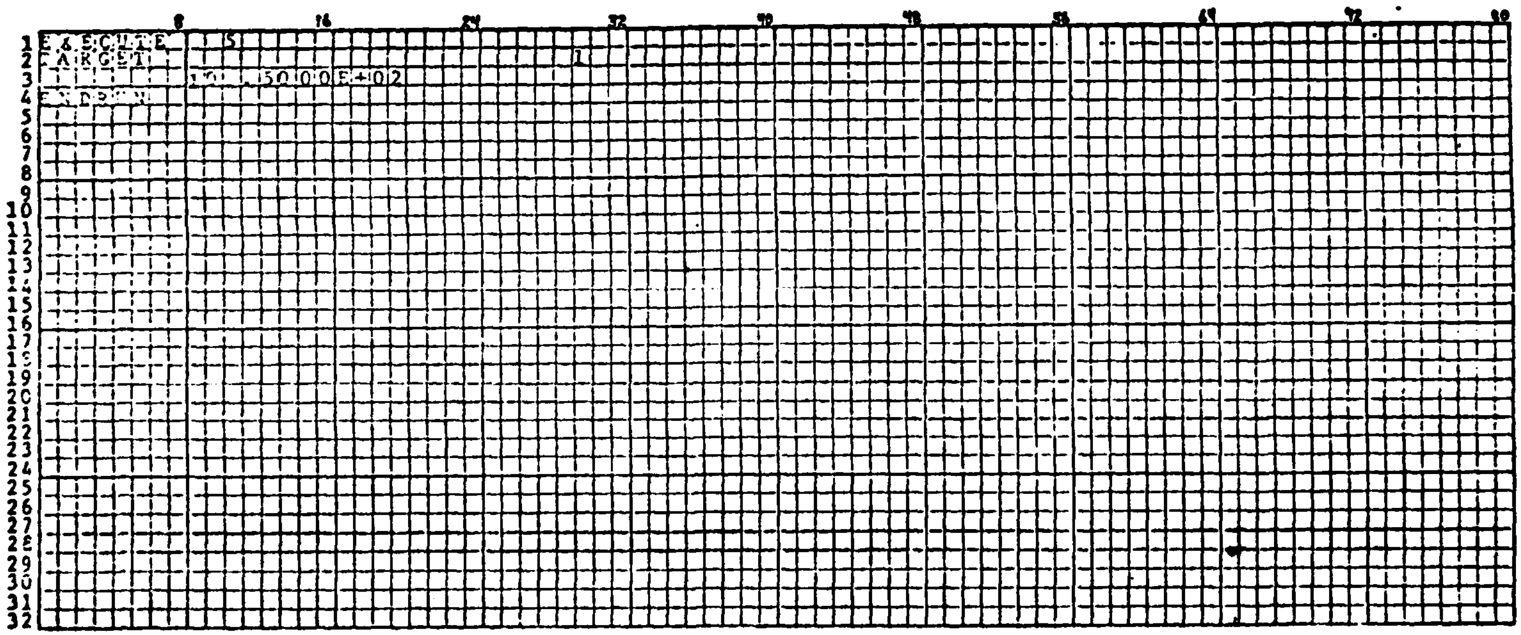


Figure 7. Execution List - Sample Problem B

SENSØR NUMBER = 5

1= VISUAL ØBSERVER
2= FØRWARD-LØØKING INFRARED
3= ACTIVE (ILLUMINATED) TV
4= PASSIVE (DAYLIGHT) TV
5= FØRWARD-LØØKING RADAR,MTI
6= FØRWARD-LØØKING RADAR,NØN-MTI
7= SYNTHETIC APERTURE RADAR

NAME	IVAR	VALUE
TARG	10	.5000E+02

I = MINIMUM LAUNCH POINT
II = DIVE BEGINS
III= (DUMMY PØSITIØN)
IV = TARGET PASSES ØUT ØF FØV(FIXED DEPRESSIØN ANGLE)
V = SEARCH ALTITUDE ACHIEVED
VI = CLIMB TØ ALTITUDE BEGINS

X = ALØNG-TRACK GRØUND DISTANCE TØ TARGET,FT.
XY = GRØUND RANGE TØ TARGET,FT.
TIME= TIME BEFØRE LAUNCH,SEC.
PLØS= PRØBABILITY TARGET IS WITHIN LØS
PFØV= PRØBABILITY TARGET IS WITHIN FØV
P2 = SEARCH TERM PRØBABILITY
P3D = DISCRIMINABILITY(DETECTIØN)
P3R = DISCRIMINABILITY(RECØGNITIØN)
PAD = CUMULATIVE PRØBABILITY ØF DETECTIØN
PAR = CUMULATIVE PRØBABILITY ØF RECØGNITIØN

I	II	III	IV	V	VI
			4768.	22689.	
			4768.	22689.	

X	XY	TIME	PLØS	PFØV	P2	P3D	P3R	PAD	PAR
4993.	4993.	.33	1.000	1.000	1.000	0.000	0.000	0.000	0.000
5668.	5668.	1.33	.999	1.000	1.0000	0.000	0.000	0.000	0.000
6342.	6342.	2.33	.998	1.000	1.000	0.000	0.000	0.000	0.000
7016.	7016.	3.33	.997	1.000	1.000	0.000	0.000	0.000	0.000
7691.	7691.	4.33	.996	1.000	1.000	0.000	0.000	0.000	0.000
8365.	8365.	5.33	.994	1.000	1.000	1.000	.604	.994	.600
9040.	9040.	6.33	.992	1.000	1.000	1.000	.604	.994	.600
9714.	9714.	7.33	.989	1.000	1.000	1.000	.603	.994	.600
10389.	10389.	8.33	.985	1.000	1.000	1.000	.603	.994	.600
11063.	11063.	9.32	.981	1.000	1.000	1.000	.603	.994	.600
11737.	11737.	10.32	.977	1.000	1.000	1.000	.603	.994	.600
12412.	12412.	11.32	.972	1.000	1.000	1.000	.603	.994	.600
13086.	13086.	12.32	.967	1.000	1.000	1.000	.603	.994	.600
13761.	13761.	13.32	.961	1.000	1.000	1.000	.603	.994	.600
14435.	14435.	14.32	.955	1.000	1.000	1.000	.603	.994	.600
15110.	15110.	15.32	.948	1.000	1.000	1.000	.603	.994	.600
15784.	15784.	16.32	.942	1.000	1.000	1.000	.603	.994	.600
16459.	16459.	17.32	.935	1.000	1.000	1.000	.603	.994	.600
17133.	17133.	18.32	.928	1.000	1.000	1.000	.603	.994	.600
17807.	17807.	19.31	.921	1.000	1.000	1.000	.603	.994	.600
18482.	18482.	20.31	.913	1.000	1.000	1.000	.603	.994	.600
19156.	19156.	21.31	.906	1.000	1.000	1.000	.603	.994	.600
19831.	19831.	22.31	.898	1.000	1.000	1.000	.603	.994	.600
20505.	20505.	23.31	.890	1.000	1.000	1.000	.603	.994	.600
21180.	21180.	24.31	.882	1.000	1.000	1.000	.603	.994	.600
21854.	21854.	25.31	.874	1.000	1.000	1.000	.603	.994	.600
22528.	22528.	26.31	.867	1.000	1.000	1.000	.603	.994	.600

Figure 8. Computer Output - Sample Problem B

C. SYNTHETIC APERTURE RADAR (SAR) - SAMPLE PROBLEM C

This sample problem exemplifies the use of the Synthetic Aperture Radar model. The search aircraft is flying at an altitude of 10,000 feet and looking at an offset of 150,000 feet.

The specific variables which were entered into the program are shown in the execution list of Figure 9. One "data name" card and two "data modification" cards were required. All other library values remain unchanged.

The results of the computer run can be seen in the computer output of Figure 10. The Synthetic Aperture Radar processes the doppler phase history of the illuminated ground area over the entire illumination period to generate the display. All returns from the target are integrated into this display. Thus, only one line of output is necessary to present the results. The radar beam was directed 26,509 feet ahead of the flight path and 152,323 feet from the aircraft. The cumulative probability of detection (PAD) achieved was .178.

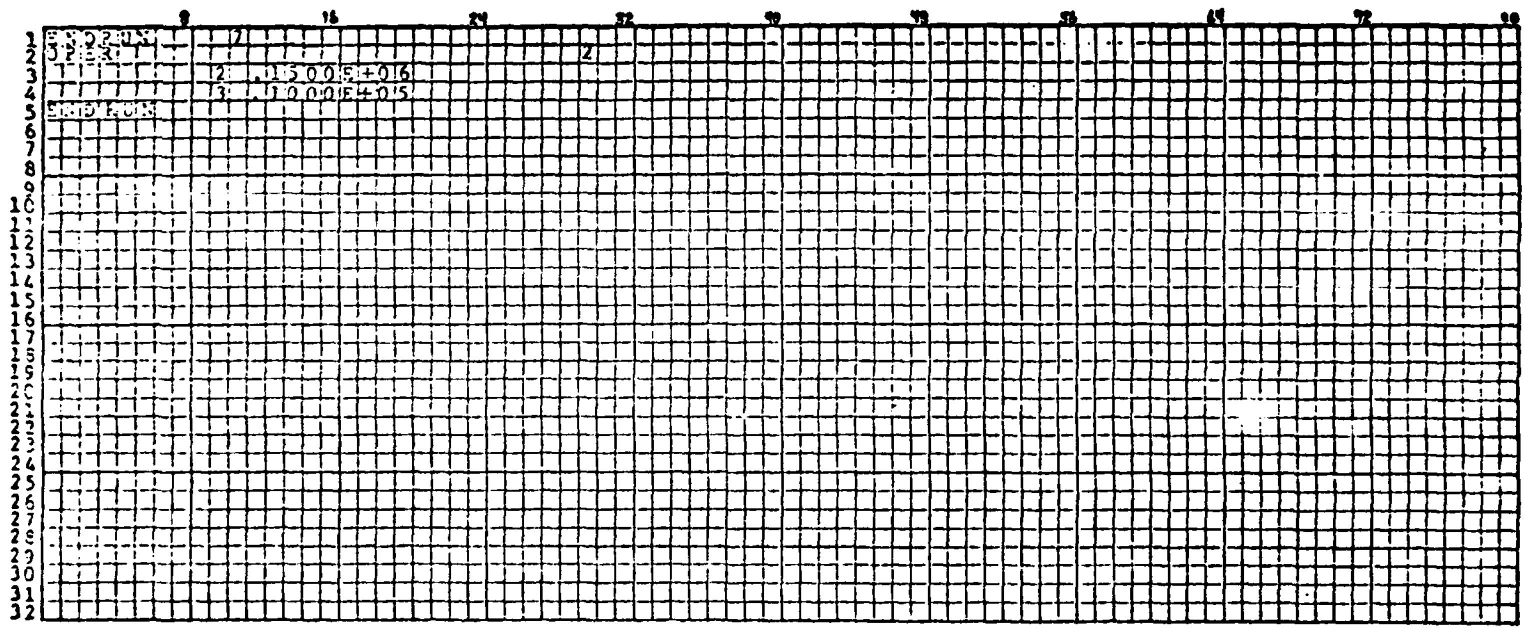


Figure 9. Execution List - Sample Problem C

```
            SENSØR NUMBER = 7
  1= VISUAL ØBSERVER
  2= FORWARD-LØØKING INFRARED
  3= ACTIVE (ILLUMINATED) TV
  4= PASSIVE (DAYLIGHT) TV
  5= FØRWARD-LØØKING RADAR,MTI
  6= FØRWARD-LØØKING RADAR,NØN-MTI
  7= SYNTHETIC APERTURE RADAR

NAME  IVAR    VALUE
ØPER    2    .1500E+06
ØPER    3    .1000E+05

X    = ALØNG-TRACK GRØUND DISTANCE TØ TARGET,FT.
XY   = GRØUND RANGE TØ TARGET,FT.
TIME= TIME BEFØRE LAUNCH,SEC.
PLØS= PRØBABILITY TARGET IS WITHIN LØS
PFØV= PRØBABILITY TARGET IS WITHIN FØV
P2   = SEARCH TERM PRØBABILITY
P3D  = DISCRIMINABILITY(DETECTIØN)
P3R  = DISCRIMINABILITY(RECØGNITIØN)
PAD  = CUMULATIVE PRØBABILITY ØF DETECTIØN
PAR  = CUMULATIVE PRØBABILITY ØF RECØGNITIØN


    X        XY      TIME   PLØS  PFØV   P2     P3D    P3R   PAD   PAR
 26509.  152323.             .508 1.000  .350 1.000   .431  .178  .077
```

Figure 10. Computer Output - Sample Problem C

VII. TATAC PROGRAM LISTINGS

This section contains the listings of the TATAC computer programs written in FORTRAN IV. The following listing includes all routines needed to run the TATAC model.

```
      PROGRAM TATAC(INPUT,OUTPUT)
      COMMON/ARRAYS/FD(250)
      COMMON/ARRSET/IC(50)
      COMMON/TABLES/FDTAB(725)
      COMMON/TABSET/ICT(50)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DELTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      DIMENSION ID(250)
      EQUIVALENCE (FD,ID)
      ISNUMB=1
      CB=.01745
      CALL INPUT1
      II0=IC(1)
      IF0=IC(2)
      IIT=IC(3)
      IFT=IC(4)
      IIB=IC(5)
      IFB=IC(6)
      IIEVT=IC(7)
      IFEVT=IC(8)
      IITS=IC(9)
      IFTS=IC(10)
      CALL INPUT2(ISNUMB)
      GO TO(10,20,30,40,50,60,70),ISNUMB
   10 ISETID=IC(11)
      ISETFD=IC(12)
      ISENS=1
      ISTYPF=1
      GO TO 200
   20 ISETID=IC(13)
      ISETFD=IC(14)
      ISENS=2
      GO TO 75
   30 ISETID=IC(15)
      ISETFD=IC(16)
      ISENS=3
      GO TO 75
   40 ISETID=IC(17)
      ISETFD=IC(18)
      ISENS=4
      GO TO 75
   50 ISETID=IC(19)
      ISETFD=IC(20)
      ISENS=5
      ISTYPF=2
      GO TO 80
   60 ISETID=IC(21)
      ISETFD=IC(22)
      ISENS=6
```

```
      ISTYPE=2
      GØ TØ 80
   70 ISFTID=IC(23)
      ISFTFD=IC(24)
      ISENS=7
      ISTYPE=2
      GØ TØ 100
   75 ISTYPE=ID(ISFTID)
      ASPECT=FD(ISFTFD)
      THETAV=FD(ISFTFD+2)*CR
      THETAH=ASPECT*THETAV
      PHID=FD(ISFTFD+3)*CR
      GØ TØ 200
   80 THETAV=FD(ISFTFD+10)*CR
      THETAH=FD(ISFTFD+8)*CR
      PHID=FD(ISFTFD+7)*CR+THETAV/2.
      CALL FLIGHT(ISENS,ISTYPE,FD(IFØ),ID(ISFTID),FD(ISFTFD))
      PRINT 202
      PRINT 203
      PRINT 204
      PRINT 85,X4,X5
      PRINT 85,XY4,XY5
   85 FØRMAT(1X,34X,2(F7.0,1X))
      PRINT 206
      TIME=0.
      IPRNT2=-1
      IPRNT1=-1
      PAD=0.
      PAR=0.
   90 CALL GEØM(ISENS,ISTYPE,FD(IFØ),ISTØP)
      IF(ISTØP.EQ.1) GØ TØ 600
      TIME=TIME+DELTAT
      PFØV=1.0
      CALL LØS(ISENS,ISTYPE,ID(IIFVT),FD(IFFVT),PLØS)
      P1=PLØS*PFØV
      CALL SENFLR(ISENS,ID(IIT),FD(IFT),FD(IFR),FD(IFFVT),
     $ FD(IFTS),ID(ISFTID),FD(ISFTFD),P2,P3D,P3R)
      IF(P3D.GT.0.) GØ TØ 380
      IPRNT1=IPRNT1+1
      IF(((IPRNT1/3)*3).NE.IPRNT1) GØ TØ 90
      PRINT 425,X,XY,TIME,PLØS,PFØV,P2,P3D,P3R,PAD,PAR
      GØ TØ 90
  100 CONTINUE
      PFØV=1.0
      CALL SAR(FD(IFØ),ID(IIT),FD(IFT),FD(IFR),FD(IFFVT),
     $ ID(ISFTID),FD(ISFTFD),P2,P3D,P3R)
      CALL LØS(ISENS,ISTYPE,ID(IIFVT),FD(IFFVT),PLØS)
      P1=PLØS*PFØV
      PAD=P1*P2*P3D
      PAR=P1*P2*P3R
```

```
      PRINT 203
      PRINT 206
      PRINT 105,X,XY,PLOS,PFOV,P2,P3D,P3R,PAD,PAR
  105 FORMAT(1X,2(F7.0,1X),6X,7(F5.3,1X))
      GO TO 600
C
  200 CALL FLIGHT(ISENS,ISTYPE,FD(IFO),ID(ISETID),FD(ISETFD))
      PRINT 202
  202 FORMAT(1X,/,1X,*I   = MINIMUM LAUNCH POINT*,
     $/,1X,*II  = DIVE BEGINS*,
     $/,1X,*III= (DUMMY POSITION)*,
     $/,1X,*IV  = TARGET PASSES OUT OF FOV(FIXED DEPRESSION*,
     $* ANGLE)*,
     $/,1X,*V   = SEARCH ALTITUDE ACHIEVED*,
     $/,1X,*VI  = CLIMB TO ALTITUDE BEGINS*)
      PRINT 203
      PRINT 204
      PRINT 205,X1,X2,X3,X4,X5,X6
      PRINT 205,XY1,XY2,XY3,XY4,XY5,XY6
  203 FORMAT(1X,/,1X,*X    = ALONG-TRACK GROUND DISTANCE TO *,
     $*TARGET,FT.*,/,1X,*XY   = GROUND RANGE TO TARGET,FT.*,
     $/,1X,*TIME= TIME BEFORE LAUNCH,SEC.*,
     $/,1X,*PLOS= PROBABILITY TARGET IS WITHIN LOS*,
     $/,1X,*PFOV= PROBABILITY TARGET IS WITHIN FOV*,
     $/,1X,*P2  = SEARCH TERM PROBABILITY*,
     $/,1X,*P3D = DISCRIMINABILITY(DETECTION)*,
     $/,1X,*P3R = DISCRIMINABILITY(RECOGNITION)*,
     $/,1X,*PAD = CUMULATIVE PROBABILITY OF DETECTION*,
     $/,1X,*PAR = CUMULATIVE PROBABILITY OF RECOGNITION*,/)
  204 FORMAT(1X,14X,*I*,6X,*II*,6X,*III*,5X,*IV*,6X,*V*,7X,*VI*)
  205 FORMAT(1X,10X,6(1X,F7.0))
      PRINT 206
  206 FORMAT(1X,/,4X,*X*,6X,*XY*,5X,*TIME*,2X,*PLOS*,2X,*PFOV*,
     $2X,*P2*,4X,*P3D*,3X,*P3R*,3X,*PAD*,3X,*PAR*)
      IPHASE=0
      TIME=0.
      IPRNT1=1
      X0=0.
      XY0=0.
      TIME0=0.
      PLOS0=0.
      PFOV0=0.
      P20=0.
      P3D0=0.
      P3R0=0.
      PAD0=PAD
      PAR0=PAR
      CALL GEOM(ISENS,ISTYPE,FD(IFO),ISTOP)
      IF(ISTOP.EQ.1) GO TO 600
      TIME=TIME+DELTAT
```

```
300 IPHASE=IPHASE+1
    IF(IPHASE.GT.2) GO TO 600
    IF(IPHASE.EQ.2) PRINT 425,X0,XY0,TIME0,PLOS0,PFOV0,
   $ P20,P3D0,P3R0,PAD0,PAR0
    IPRNT2=-1
    IF(ISENS.EQ.1) GO TO 305
    IFOV=IPHASE
    ASPECT=FD(ISETFD)
    THETAV=FD(ISETFD+IFOV)*CR
    THETAH=ASPECT*THETAV
    ISTYPE=ID(ISETID)
305 CONTINUE
    CALL SEARCH(ISENS,ISTYPE,ID(IIT),FD(IFT),
   $ FD(IFEVT),FD(IFTS),P2)
    CALL FOV(ISENS,ISTYPE,FD(IFTS),PFOV)
    CALL LOS(ISENS,ISTYPE,ID(IIEVT),FD(IFEVT),PLOS)
    P1=PLOS*PFOV
    GO TO(310,320,330,330,340,340,350),ISNUMB
310 CALL VISOB(FD(IFT),FD(IFB),ID(IIEVT),FD(IFEVT),
   $ ID(ISETID),FD(ISETFD),P3D,P3R)
    GO TO 380
320 CALL FLIR(FD(IFT),FD(IFB),FD(IFEVT),
   $ ID(ISETID),FD(ISETFD),P3D,P3R)
    GO TO 380
330 CALL TV(ISENS,FD(IFT),FD(IFB),ID(IIEVT),FD(IFEVT),
   $ ID(ISETID),FD(ISETFD),IFOV,P3D,P3R)
    GO TO 380
340 CALL SENFLR(ISENS,ID(IIT),FD(IFT),FD(IFB),FD(IFEVT),
   $ FD(IFTS),ID(ISETID),FD(ISETFD),P2,P3D,P3R)
    GO TO 380
350 CONTINUE
    GO TO 100
380 OR=P1*P2
    PAD=P3D*OR
    PAR=P3R*OR
400 IF(ISENS.GT.4) GO TO 410
    IF((X.LE.X2).AND.(IPRNT1.EQ.0)) GO TO 450
    IF((X.GT.X2).AND.(P3D.LT..005)) GO TO 600
    GO TO 420
410 IF((P1.OR.P2.OR.P3D).LT..005) GO TO 600
420 IPRNT2=IPRNT2+1
    IF(((IPRNT2/3)*3).NE.IPRNT2) GO TO 450
    PRINT 425,X,XY,TIME,PLOS,PFOV,P2,P3D,P3R,PAD,PAR
425 FORMAT(1X,2(F7.0,1X),F5.2,7(1X,F5.3))
    IPRNT1=0
450 CONTINUE
    ORN1=OR
    PADN1=PAD
    PARN1=PAR
    CALL GEOM(ISENS,ISTYPE,FD(IFG),ISTOP)
```

```
      IF(ISTOP.EQ.1) GO TO 600
      TIME=TIME+DELTAT
      IF(ISENS.GT.4) GO TO 460
      IF((X.GT.X2).AND.(IPHASE.EQ.1)) GO TO 300
      CALL SEARCH(ISENS,ISTYPE,ID(IIT),FD(IFT),
     $ FD(IFFVT),FD(IFTS),P2)
      CALL FOV(ISENS,ISTYPE,FD(IFTS),PFOV)
  460 CALL LOS(ISENS,ISTYPE,ID(IIFVT),FD(IFFVT),PLOS)
      P1=PLOS*PFOV
      GO TO(510,520,530,530,540,540,550),ISNUMB
  510 CALL VISOB(FD(IFT),FD(IFB),ID(IIEVT),FD(IFEVT),
     $ ID(ISETID),FD(ISETFD),P3D,P3R)
      GO TO 580
  520 CALL FLIR(FD(IFT),FD(IFB),FD(IFEVT),
     $ ID(ISETID),FD(ISETFD),P3D,P3R)
      GO TO 580
  530 CALL TV(ISENS,FD(IFT),FD(IFB),ID(IIFVT),FD(IFEVT),
     $ ID(ISETID),FD(ISETFD),IFOV,P3D,P3R)
      GO TO 580
  540 CALL SENFLR(ISENS,ID(IIT),FD(IFT),FD(IFB),FD(IFEVT),
     $ FD(IFTS),ID(ISETID),FD(ISETFD),P2,P3D,P3R)
      GO TO 580
  550 CONTINUE
      GO TO 100
  580 QR=P1*P2+(1.-P2)*QRN1
      X0=X
      XY0=XY
      TIME0=TIME
      PLOS0=PLOS
      PFOV0=PFOV
      P20=P2
      P3D0=P3D
      P3R0=P3R
      DELTQR=QR-QRN1
      IF(DELTQR.LE.0.) DELTQR=0.
      PAD=PADN1+DELTQR*P3D
      PAR=PARN1+DELTQR*P3R
      PAD0=PAD
      PAR0=PAR
      GO TO 400
  600 CONTINUE
      END
      SUBROUTINE INPUT1
      COMMON/ARRAYS/FD(250)
      COMMON/ARRSET/IC(50)
      COMMON/TABLES/FDTAB(725)
      COMMON/TABSET/ICT(50)
      DIMENSION ID(250)
      EQUIVALENCE (FD,ID)
      DATA(IC(I),I=1,24)/1,2,16,19,36,37,43,48,56,57,
```

```
$ 66,68,73,75,85,87,111,113,136,138,161,163,186,188/
 DATA(IC(I),I=25,50)/26*211/
 DATA(ICT(I),I=1,12)/1,44,0,89,4,0,97,4,0,105,3,3/
 DATA(ICT(I),I=13,42)/120,3,7,151,2,4,165,4,9,
$ 214,4,9,263,4,7,302,4,7,341,6,25,522,6,25,
$ 703,4,0,711,4,0/
 DATA(ICT(I),I=43,50)/8*719/
 DATA ID(1),ID(16),ID(17),ID(18),ID(36)/0,10,0,0,0/
 DATA ID(43),ID(44),ID(45),ID(46),ID(47)/2,2,0,0,0/
 DATA ID(56),ID(66),ID(67),ID(73),ID(74)/0,1,0,1,0/
 DATA ID(85),ID(86),ID(111),ID(112),ID(136)/1,0,1,0,1/
 DATA ID(137),ID(161),ID(162),ID(186),ID(187)/0,1,0,1,0/
 DATA(FD(I),I=2,15)/0.,4000.,500.,400.,400.,400.,
$ 400.,3100.,60000.,27.,10.,.333,2*0./
 DATA(FD(I),I=19,35)/20.5,10.7,8.8,.16,306.,.9,11.7,
$ 11.7,11.7,90.,0.,25.,5*0./
 DATA(FD(I),I=37,42)/.08,300.,.91,0.,6.,0./
 DATA(FD(I),I=48,55)/.1,5.,0.,6.,.4,4.,.25,0./
 DATA(FD(I),I=57,65)/0.,36000.,100.,20.,255.,255.,3*0./
 DATA(FD(I),I=68,72)/36000.,0.,0.,.55,0./
 DATA(FD(I),I=75,84)/1.333,1.86,7.44,30.,.0306,
$ .01039,4*0./
 DATA(FD(I),I=87,110)/1.,1.,1.,30.,.45E7,25.,5.6,5.6,
$ 1.,.6,.45E-2,.2E-4,.2E-1,.2E-8,.84,400.,511.,
$ 50.,.24,1.,4*0./
 DATA(FD(I),I=113,135)/1.,1.,4.,30.,.925E7,15.7,5.6,5.6,
$ 1.,.6,.45E-2,.8E-8,.8E-5,.7E-8,.55,600.,511.,0.,
$ 0.,1.,3*0./
 DATA(FD(I),I=138,160)/24.,1000.,.2E-12,3600.,60000.,.03,
$ 90.,10.,2.5,90.,30.,.2E-5,.5,.8,0.,0.,0.,0.,.95,
$ 0.,15.,0.,8./
 DATA(FD(I),I=163,185)/24.,1000.,.2E-12,3600.,60000.,.03,
$ 90.,10.,2.5,90.,30.,.2E-5,.5,.8,0.,0.,0.,0.,.95,
$ 0.,15.,0.,8./
 DATA(FD(I),I=188,210)/6.,1000.,.2E-12,3600.,60000.,50.,
$ 15.,0.,.03,0.,0.,30.,.1E-5,.5,.8,0.,1.,10.,.95,
$ 36.,36.,36.,1./
 DATA(FD(I),I=211,250)/40*0./
 DATA(FDTAB(I),I=1,44)/
$ 0.00,0.05,0.10,0.15,0.20,0.25,0.30,0.35,0.40,0.45,
$ 0.50,0.55,0.60,0.65,0.70,0.75,0.80,0.85,0.90,0.95,
$ 1.00,1.05,1.10,1.15,1.20,1.25,1.30,1.35,1.40,1.45,
$ 1.50,1.55,1.60,1.65,1.70,1.75,1.80,1.85,1.90,1.95,
$ 2.00,2.50,4.00,5.00/
 DATA(FDTAB(I),I=45,88)/
$ 1.0000000,0.8278345,0.7225450,0.6410387,0.5742006,
$ 0.5177301,0.4691153,0.4267127,0.3893680,0.3562290,
$ 0.3266439,0.3000996,0.2761839,0.2545597,0.2349471,
$ 0.2171109,0.2008517,0.1859986,0.1724041,0.1599404,
$ 0.1484955,0.1379713,0.1282811,0.1193481,0.1111041,
```

```
$ 0.1034881,0.0964455,0.0899275,0.0838899,0.0782930,
$ 0.0731008,0.0682807,0.0638032,0.0596413,0.0557706,
$ 0.0521687,0.0488153,0.0456915,0.0427803,0.0400660,
$ 0.0375343,0.0197976,0.0031982,0.0009964/
 DATA(FDTAB(I),I=89,104)/100.,120.,140.,180.,.3,.405,
$ .4675,.5,100.,120.,150.,180.,.513,.633,.693,.7/
 DATA(FDTAB(I),I=105,119)/5.,40.,75.,1.,2.,3.,-43.,-29.,
$ -25.,-39.,-25.,-20.,-23.,-17.,-10./
 DATA(FDTAB(I),I=120,150)/.032,.018,.0086,0.,.25,.51,
$ 2.54,6.35,12.70,25.40,0.,1.4E-6,4.2E-6,2.7E-5,9.2E-5,
$ 2.2E-4,5.6E-4,0.,8.6E-6,2.0E-5,1.4E-4,3.5E-4,9.0E-4,
$ 1.9E-3,0.,6.2E-5,1.1E-4,8.5E-4,1.6E-3,4.3E-3,7.6E-3/
 DATA(FDTAB(I),I=151,164)/2.,1.,1.,2.,3.,4.,1823.,3414.,
$ 4816.,4333.,366.,914.,1829.,1524./
 DATA(FDTAB(I),I=165,213)/1.,2.,3.,4.,0.,.125,.25,.375,.5,
$ .625,.75,.875,1.,0.,3*.25,.4,4*.5,0.,3*.2,2*.4,3*.6,
$ 0.,2*.2,2*.4,2*.6,.8,1.,0.,3*.2,2*.4,3*.6/
 DATA(FDTAB(I),I=214,262)/1.,2.,3.,4.,0.,.125,.25,.375,.5,
$ .625,.75,.875,1.,.141,.061,.039,.038,.028,.038,.053,
$ .140,.462,.226,.120,.057,.078,.050,.067,.091,.105,.207,
$ .154,.177,.087,.092,.071,.085,.095,.149,.09,.216,.113,
$ .052,.052,.046,.048,.062,.125,.286/
 DATA(FDTAB(I),I=263,301)/1.,2.,3.,4.,0.,.25,.51,
$ 2.54,6.35,12.70,25.40,.924,.056,.02,4*0.,.929,.048,
$ .021,.002,3*0.,.8937,.058,.039,.008,.001,.0003,0.,.889,
$ .064,.042,.005,3*0./
 DATA(FDTAB(I),I=302,340)/1.,2.,3.,4.,0.,.25,.51,
$ 2.54,6.35,12.70,25.4,.9811,.0078,.0064,.0030,.0014,
$ .0003,0.,.9794,.0079,.0068,.0037,.0018,.0004,0.,.9771,
$ .0068,.0066,.0055,.0032,.0008,0.,.97762,.0069,.0068,
$ .0051,.0029,.00068,0./
 DATA(FDTAB(I),I=341,430)/1000.,1500.,5000.,10000.,
$ 20000.,30000.,1.,1.5,2.,3.,4.,5.,6.,7.,8.,9.,10.,
$ 15.,20.,25.,30.,35.,40.,50.,60.,70.,80.,90.,100.,
$ 150.,200.,.02,.02,.03,.06,.08,.12,.16,.20,.24,.30,
$ .35,.73,1.35,2.25,3.5,5.4,8.4,21.5,90.,6*90.,
$ 0.,.02,.03,.06,.09,.13,.17,.22,.28,.34,.42,.88,1.65,
$ 3.,5.6,10.5,23.,80.,7*80.,5*0.,.08,.1,.13,.16/
 DATA(FDTAB(I),I=431,521)/.20,.24,.51,.94,1.55,2.5,
$ 3.8,5.8,13.,38.5,100.,5*100.,10*0.,.09,.18,.31,.49,
$ .72,1.,1.36,2.35,3.85,6.3,10.,16.,25.5,100.,100.,
$ 12*0.,.14,.20,.27,.37,.50,.84,1.3,1.92,2.75,3.75,
$ 5.2,21.,100.,14*0.,.22,.27,.34,.56,.90,1.38,2.05,
$ 2.88,3.95,12.,26./
 DATA(FDTAB(I),I=522,611)/1000.,1500.,5000.,10000.,
$ 20000.,30000.,1.,1.5,2.,3.,4.,5.,6.,7.,8.,9.,10.,
$ 15.,20.,25.,30.,35.,40.,50.,60.,70.,80.,90.,100.,
$ 150.,200.,.01,.03,.04,.09,.16,.24,.33,.43,.53,.65,
$ .78,1.7,3.1,5.2,8.2,12.25,17.75,38.,84.,100.,5*100.,
$ 0.,.02,.04,.08,.15,.22,.32,.44,.56,.7,.88,2.07,4.1,
```

```
$ 7.4,12.5,20.5,32.,72.,100.,6*100.,5*0.,.07,.11,.15,
$ .19/
 DATA(FDTAB(I),I=612,702)/.25,.32,.8,1.6,2.85,4.8,
$ 7.6,12.25,30.,66.,100.,5*100.,10*0.,.1,.21,.4,.67,
$ 1.05,1.55,2.23,4.1,7.,11.2,18.2,31.5,60.,120.,120.,
$ 12*0.,.14,.22,.34,.46,.64,1.1,1.75,2.6,3.9,5.4,7.6,
$ 34.,160.,14*0.,.22,.28,.36,.58,.89,1.3,1.9,2.7,3.65,
$ 15.,47./
 DATA(FDTAB(I),I=703,710)/95.,120.,150.,180.,
$ 1.95,1.1,.65,1./
 DATA(FDTAB(I),I=711,718)/95.,120.,150.,180.,
$ 2.78,1.78,1.1,1./
 DATA(FDTAB(I),I=719,725)/7*0./
 RETURN
 END
 SUBROUTINE INPUT2(ISNUMB)
 COMMON/ARRAYS/FD(250)
 COMMON/ARRSET/IC(50)
 DIMENSION ID(250),XTEMP(10),ITEMP1(10),ITEMP2(10,10),
$TEMP3(10,10)
 EQUIVALENCE (FD,ID)
 DATA XEXEC,XENDR,XOPER,XTARG/4HEXEC,4HENDR,4HOPER,4HTARG/
 DATA XBACK,XENVI,XSEAR,XSENS/4HBACK,4HENVI,4HSEAR,4HSENS/
 DATA XBLANK,XTERR/4H    ,4HTERR/
 ISNUMB=1
 NTEMP1=0
 DO 1 J=1,10
1 XTEMP(J)=XBLANK
 READ 5,AAA,KSNUMB
5 FORMAT(A4,6X,I1)
 IF(AAA.EQ.XEXEC) GO TO 15
 PRINT 10
10 FORMAT(* EXECUTE CARD MISSING OR OUT OF ORDER*)
 GO TO 30
15 IF((KSNUMB.GT.0).AND.(KSNUMB.LT.8)) GO TO 25
 PRINT 20,KSNUMB
20 FORMAT(* INVALID SENSOR NUMBER--*,I2)
 GO TO 30
25 ISNUMB=KSNUMB
30 ISET=5+ISNUMB
35 READ 40,XNAME,IN
40 FORMAT(A4,24X,I2)
 II=0
 IF(XNAME.EQ.XENDR) GO TO 200
 IF(XNAME.EQ.XOPER) II=1
 IF(XNAME.EQ.XTARG) II=2
 IF(XNAME.EQ.XBACK) II=3
 IF(XNAME.EQ.XENVI) II=4
 IF(XNAME.EQ.XTERR) II=4
 IF(XNAME.EQ.XSEAR) II=5
```

```
      IF(XNAME.EQ.XSENS) II=ISFT
      IF(II.NE.0) GO TO 50
      PRINT 45,XNAME
   45 FORMAT(* INVALID DATA NAME--*,A4)
      GO TO 35
   50 ILOOP=0
      NTEMP1=NTEMP1+1
      XTEMP(NTEMP1)=XNAME
      ITEMP1(NTEMP1)=IN
      NTEMP2=0
      I1=2*II-1
      I2=2*II
      I3=2*II+1
      II1=IC(I1)-1
      II2=IC(I2)-1
      II3=IC(I3)
   60 ILOOP=ILOOP+1
      IF(ILOOP.GT.IN) GO TO 35
      READ 70,IVAR,VALUE
   70 FORMAT(8X,I2,F10.4)
      IF(IVAR.GT.0) GO TO 80
      PRINT 75,IVAR
   75 FORMAT(* VARIABLE NUMBER OUTSIDE RANGE OF DATA NAME*,
     $* ARRAY--*,I2)
      GO TO 60
   80 IF(IVAR.LE.(II3-II2)) GO TO 100
      PRINT 90,IVAR
   90 FORMAT(* VARIABLE NUMBER OUTSIDE RANGE OF DATA NAME*,
     $* ARRAY--*,I2)
      GO TO 60
  100 JJ=II1+IVAR
      NTEMP2=NTEMP2+1
      ITEMP2(NTEMP1,NTEMP2)=IVAR
      TEMP3(NTEMP1,NTEMP2)=VALUE
      IF(IVAR.GT.(II2-II1)) GO TO 110
      ID(JJ)=IFIX(VALUE)
      GO TO 60
  110 FD(JJ)=VALUE
      GO TO 60
  200 PRINT 205,ISNUMB
  205 FORMAT(1X,/,10X,*SENSOR NUMBER = *,I1,
     $/,1X,* 1= VISUAL OBSERVER*,
     $/,1X,* 2= FORWARD-LOOKING INFRARED*,
     $/,1X,* 3= ACTIVE (ILLUMINATED) TV*,
     $/,1X,* 4= PASSIVE (DAYLIGHT) TV*,
     $/,1X,* 5= FORWARD-LOOKING RADAR,MTI*,
     $/,1X,* 6= FORWARD-LOOKING RADAR,NON-MTI*,
     $/,1X,* 7= SYNTHETIC APERTURE RADAR*)
      LOOP=0
  210 LOOP=LOOP+1
```

```
      IF(LOOP.GT.10) GO TO 240
      IF(XTEMP(LOOP).NE.XBLANK) GO TO 215
      RETURN
  215 IF(LOOP.EQ.1) PRINT 220
  220 FORMAT(1X,/,1X,*NAME   IVAR     VALUE*)
      NN=ITEMP1(LOOP)
      DO 230 J=1,NN
      PRINT 225,XTEMP(LOOP),ITEMP2(LOOP,J),TEMP3(LOOP,J)
  225 FORMAT(1X,A4,3X,I2,3X,E10.4)
  230 CONTINUE
      GO TO 210
  240 PRINT 245
  245 FORMAT(* MORE MODIFICATIONS THAN LISTED*)
      RETURN
      END
      SUBROUTINE FLIGHT(ISENS,ISTYPE,FDO,IDS,FDS)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK2/HL,HP,SPD,SPL,SPC,SPP
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLD,ANGLT,DELTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      DIMENSION FDO(20),IDS(20),FDS(20)
      C8=.01745
      Y=FDO(1)
      HL=FDO(2)
      SPL=FDO(5)*1.687778
      IF(ISENS.GT.4) GO TO 500
      ANGLD=FDO(10)*C8
      SPD=FDO(4)*1.687778*COS(ANGLD)
      SR1=FDO(8)
      DELTAT=FDO(12)
      XY2=HL/TAN(ANGLD)
      X2=SQRT(XY2*XY2-Y*Y)
      TEMP1=ATAN(Y/X2)
      XY1=SR1*COS(ANGLD)
      X1=XY1*COS(TEMP1)
      IF(ISENS.EQ.1) GO TO 300
      IF(ISTYPE.EQ.2) GO TO 200
  100 PHID=ANGLD
      HP=FDO(3)
      SPP=FDO(7)*1.687778
      ANGLC=FDO(11)*C8
      SPC=FDO(6)*1.687778*COS(ANGLC)
      SR2=FDO(9)
      XY3=XY2
      X3=X2
      XY4=XY2
      X4=X2
      XY5=SQRT(SR2*SR2-HL*HL)
      X5=SQRT(XY5*XY5-Y*Y)
      X6=X5+(HL-HP)/TAN(ANGLC)
```

```
      XY6=SORT(X6*X6+Y*Y)
      GO TO 400
  200 CALL COVER(ISENS,IDS,FDS,Y,HL,XMAX,XMIN)
      IF(XMAX.GE.X2) GO TO 210
      PHID=(ATAN(HL/X2)+THETAV/2.)/CR
      PRINT 205,PHID
  205 FORMAT(* DEPRESSION ANGLE TOO STEEP. PHID RESET= *,
     $       F5.2,* DEGREES *)
      PHID=PHID*CR
      CALL COVER(ISENS,IDS,FDS,Y,HL,XMAX,XMIN)
  210 X5=XMAX
      XY5=SORT(X5*X5+Y*Y)
      SR2=SORT(HL*HL+Y*Y+XY5*XY5)
      X6=X5
      XY6=XY5
      X3=X2
      XY3=XY2
      X4=AMAX1(X2,XMIN)
      XY4=SORT(X4*X4+Y*Y)
      HP=HL
      SPP=SPL
      SPC=SPL
      ANGLC=0.
      GO TO 400
  300 CALL COVER(ISENS,IDS,FDS,Y,HL,XMAX,XMIN)
      GO TO 210
  400 X=X1
      XY=XY1
      SR=SR1
      ANGLO=ACOS(X2/XY2)
      ANGLT=ANGLD
      H=XY*TAN(ANGLT)
      SP=SPD
      RETURN
  500 IF(ISENS.EQ.7) GO TO 600
      CALL COVER(ISENS,IDS,FDS,Y,HL,XMAX,XMIN)
      X4=XMIN
      XY4=SORT(X4*X4+Y*Y)
      X5=XMAX
      XY5=SORT(X5*X5+Y*Y)
      GO TO 700
  600 CONTINUE
  700PX=X4
      XY=XY4
      H=HL
      ANGLO=ATAN(Y/X4)
      ANGLT=ATAN(H/XY)
      SR=SORT(XY*XY+H*H)
      SP=SPL
      DELTAT=FDO(12)
```

```
      RETURN
      END
      SUBROUTINE COVER(ISENS,IDS,FDS,Y,H,XMAX,XMIN)
      EXTERNAL FUN1
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK5/X2MAX,X1MIN,VISMAX
      DIMENSION IDS(20),FDS(20)
      C8=.01745
      IF(ISENS.EQ.1) GO TO 200
      XMAX=H/TAN(PHID-THETAV/2.)
      XMIN=H/TAN(PHID+THETAV/2.)
      TEMP1=TAN(THETAH/2.)
      YMIN=XMIN*TEMP1
      IF(Y.GT.YMIN) GO TO 120
      RETURN
  120 YMAX=XMAX*TEMP1
      IF(Y.GT.YMAX) GO TO 150
      XMIN=XMAX-(YMAX-Y)/TEMP1
      RETURN
  150 Y=0.
      PRINT 155,Y
  155 FORMAT(* OFFSET(Y) GREATER THAN YMAX. SET Y= *,E10.4)
      RETURN
  200 VISMAX=FDS(1)
      RMAX=VISMAX
      ICMASK=IDS(1)
      IF(Y) 210,220,230
  210 Y=ABS(Y)
      PRINT 215
  215 FORMAT(* ONLY POSITIVE OFFSETS CONSIDERED, USED ABS. VALUE*)
      GO TO 230
  220 XMAX=RMAX
      TEMP1=FUN1(0.)*C8
      XMIN=H/TAN(TEMP1)
      IF(ICMASK.EQ.0.) XMIN=0.
      GO TO 300
  230 IF(Y.LT.RMAX) GO TO 250
      XMAX=0.
      XMIN=0.
      GO TO 300
  250 XMAX=SQRT(RMAX*RMAX-Y*Y)
      XMIN=0.
      IF(ICMASK.EQ.0) GO TO 300
      GUESS=ASIN(Y/RMAX)/C8
      TEMP1=H/TAN(FUN1(GUESS)*C8)
      TEMP2=Y/SIN(GUESS*C8)
      IF(TEMP1.LE.TEMP2) GO TO 260
      XMAX=0.
      XMIN=0.
      GO TO 300
```

```
260 GUESS=GUESS+5.
    IF(GUESS.GT.90.) GUESS=90.
    TEMP1=H/TAN(FUN1(GUESS)*CR)
    TEMP2=Y/SIN(GUESS*CR)
    IF(TEMP1.GT.TEMP2) GØ TØ 270
    IF(GUESS.LT.90.) GØ TØ 260
    XMIN=0.
    GØ TØ 300
270 CALL APPRØX(GUESS,THETAD,IFAIL)
    TEMP1=FUN1(THETAD)*CR
    TEMP1=H/TAN(TEMP1)
    XMIN=SQRT(TEMP1*TEMP1-Y*Y)
300 X2MAX=XMAX
    X1MIN=XMIN
    RETURN
    END
    SUBRØUTINE APPRØX(GUESS,ANS,IFAIL)
    EXTERNAL FUNF,FUNDF
    IFAIL=0
    ANS=GUESS
    DØ 100 I=1,50
    TEMP1=ANS
    ANS=ANS-(FUNF(ANS)/FUNDF(ANS))
    TEMP2=ABS(ANS-TEMP1)
    IF(TEMP2.LE.1.E-4) RETURN
100 CØNTINUE
    IFAIL=1
    RETURN
    END
    FUNCTIØN FUNF(THETAD)
    EXTERNAL FUN1
    CØMMØN/BLØCK2/HL,HP,SPD,SPL,SPC,SPP
    CØMMØN/BLØCK3/X,Y,XY,H,SR,SP,ANGLØ,ANGLT,DELTAT
    CR=.01745
    THETAR=THETAD*CR
    TEMP1=FUN1(THETAD)*CR
    FUNF=HL/TAN(TEMP1)-Y/SIN(THETAR)
    END
    FUNCTIØN FUNDF(THETAD)
    EXTERNAL FUN1,FUN2
    CØMMØN/BLØCK2/HL,HP,SPD,SPL,SPC,SPP
    CØMMØN/BLØCK3/X,Y,XY,H,SR,SP,ANGLØ,ANGLT,DELTAT
    CR=.01745
    THETAR=THETAD*CR
    TEMP1=FUN1(THETAD)*CR
    FUNDF=-HL*FUN2(THETAD)*CR/SIN(TEMP1)**2
   $      +HL*CØS(THETAR)/SIN(THETAR)**2
    END
    FUNCTIØN FUN1(DEGREE)
    FUN1=-.003142*DEGREE*DEGREE+.42309*DEGREE+12.9176
```

```
      END
      FUNCTION FUN2(DEGREE)
      FTN2=-.006284*DEGREE+.42309
      END
      SUBROUTINE GEOM(ISENS,ISTYPE,FDO,ISTOP)
      COMMON/BLOCK2/HL,HP,SPD,SPL,SPC,SPP
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DELTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      DIMENSION FDO(20)
      ISTOP=0
      C8=.01745
      IF(ISENS.GT.4) GO TO 200
      ANGLD=FDO(10)*C8
      IF(XY.GT.XY2) GO TO 130
      XY=XY+SPD*DELTAT
      TEMP1=ATAN(Y/X2)
      X=XY*COS(TEMP1)
      H=XY*TAN(ANGLT)
      IF(XY.LE.XY2) GO TO 120
      TGO=(XY-XY2)/SPD
      X=X2+SPL*TGO
      SP=SPL
      H=HL
  110 XY=SQRT(X*X+Y*Y)
      ANGLO=ATAN(Y/X)
      ANGLT=ATAN(H/XY)
  120 SR=SQRT(XY*XY+H*H)
      RETURN
  130 IF((ISENS.EQ.1).OR.(ISTYPE.EQ.2)) GO TO 200
      ANGLC=FDO(6)*C8
      IF(X.GT.X5) GO TO 140
      X=X+SPL*DELTAT
      IF(X.LE.X5) GO TO 110
      TGO=(X-X5)/SPL
      X=X5+SPC*TGO
      SP=SPC
      H=HL-SPC*TGO*TAN(ANGLC)
      IF(X.LE.X6) GO TO 110
      TGO=(X-X6)/SPC
      X=X6+SPP*TGO
      SP=SPP
      H=HP
      GO TO 110
  140 IF(X.GT.X6) GO TO 150
      X=X+SPC*DELTAT
      SP=SPC
      H=H-SPC*DELTAT*TAN(ANGLC)
      IF(X.LE.X6) GO TO 110
      TGO=(X-X6)/SPC
      X=X6+SPP*TGO
```

```
      SP=SPP
      H=HP
      GO TO 110
  150 X=X+SPP*DELTAT
      GO TO 110
  200 X=X+SPL*DELTAT
      IF(X.LE.X5) GO TO 110
      ISTOP=1
      GO TO 110
      END
      SUBROUTINE SEARCH(ISENS,ISTYPE,IDT,FDT,
     $FDEVT,FDTS,P2)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DELTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      COMMON/BLOCK5/X2MAX,X1MIN,VISMAX
      COMMON/BLOCK6/ALPHAH,ALPHAL,ALPHAV
      DIMENSION IDT(20),FDT(20),FDEVT(20),FDTS(20)
      F(D1,D2)=2.*ATAN((-D1-1.+SQRT((D1+1.)**2+D1*D2*D2))/(D1*D2))
      CPI=3.14159
      DTX=FDT(1)
      DTY=FDT(2)
      DTZ=FDT(3)
      NT=IDT(1)
      TSPACE=FDT(10)
      DSX=FDTS(2)
      DSY=FDTS(3)
      ANGTC=CPI/2.-ANGLT
      TANGTC=TAN(ANGTC)
      IF(ANGLO.LE.(ATAN(DTY/DTX))) GO TO 10
      TEMP1=SIN(ANGLO)
      DTXX=DTY/TEMP1
      GO TO 20
   10 TEMP1=COS(ANGLO)
      DTXX=DTX/TEMP1
   20 TEMP1=TANGTC*TANGTC
      TEMP2=(DTXX+DTZ/TAN(ANGLT))/H
      ALPHAV=F(TEMP1,TEMP2)
      ALPHAH=2.*ATAN((DTX*SIN(ANGLO)+DTY*COS(ANGLO))/(2.*SR))
      ALPHAL=2.*ATAN((DTX*COS(ANGLO)*SIN(ANGLT)+
     $      DTY*SIN(ANGLO)*SIN(ANGLT)+DTZ*COS(ANGLT))/(2.*SR))
      IF(X.LT.X4) GO TO 250
      AT=DTX*DTY*SIN(ANGLT)+
     $   (DTX*DTZ*SIN(ANGLO)+DTY*DTZ*COS(ANGLO))*COS(ANGLT)
      AT=AT*(NT+(NT-1)*TSPACE/DTX)
      IF(ISENS.EQ.1) GO TO 100
      IF(ISTYPE.EQ.1) PHID=ANGLT
      PHIC=CPI/2.-PHID
      TPHIC=TAN(PHIC)
      IF(DSX.LE.0.) GO TO 30
```

```
      TEMP1=TPHIC*TPHIC
      TEMP2=DSX/H
      BETAV=F(TEMP1,TEMP2)
      IF(BETAV.LT.THETAV) GO TO 40
   30 BETAV=THETAV
   40 IF(DSY.LE.0.) GO TO 50
      BETAH=2.*ATAN(DSY/(2.*SR))
      IF(BETAH.LT.THETAH) GO TO 60
   50 BETAH=THETAH
   60 OMEGA=BETAH*(SIN(PHID+BETAV/2.)-SIN(PHID-BETAV/2.))
      SRPHID=H/SIN(PHID)
      AS=OMEGA*SRPHID*SRPHID
      GO TO 200
  100 WLOC=FDTS(4)
      TEMP1=X2MAX+SQRT(X2MAX*X2MAX+Y*Y+H*H)
      TEMP2=X1MIN+SQRT(X1MIN*X1MIN+Y*Y+H*H)
      AS=H*WLOC*ALOG(TEMP1/TEMP2)
  200 ARATIO=AT/AS
      G=FDEVT(6)
      TEMP1=(700./G)*ARATIO*DELTAT
      IF(TEMP1.LT.180.) GO TO 270
  250 P2=1.0
      RETURN
  270 P2=1.-EXP(-TEMP1)
      RETURN
      END
      SUBROUTINE LOS(ISENS,ISTYPE,IDEVT,FDEVT,PLOS)
      COMMON/TABLES/FDTAB(725)
      COMMON/TABSET/ICT(50)
      COMMON/BLOCK3/X,Y,XY,H,SP,SP,ANGLO,ANGLT,DELTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      DIMENSION IDEVT(20),FDEVT(20)
      IF(X.LT.X4) GO TO 35
      CR=.01745
      ICFLOS=IDEVT(3)
      ANGLM=FDEVT(2)*CR
      RATIOM=FDEVT(3)
      PNMASK=1.0
      IF(ASIN(H/SR).GT.ATAN(RATIOM)) GO TO 10
      PLOS=0.
      RETURN
   10 IF(ANGLM.LE.0.) GO TO 20
      PNMASK=PNMASK*(1.-EXP(-ANGLT/ANGLM))
   20 PCFLOS=1.0
      IF(ISENS.GT.4) GO TO 30
      IF(ICFLOS.EQ.0) GO TO 30
      ZENITH=ANGLT/CR+90.
      JJ=ICT(4)
      NX=ICT(5)
      IX=JJ
```

```
      IY=IX+NX
      IF(H.LE.5000.) GO TO 25
      JJ=ICT(7)
      NX=ICT(8)
      IX=JJ
      IY=IX+NX
   25 CALL INTERI(FDTAB(IX),FDTAB(IY),NX,ZENITH,PCFLOS)
   30 PLOS=PNMASK*PCFLOS
      RETURN
   35 PLOS=1.0
      RETURN
      END
      SUBROUTINE INTERI(XX,YY,NN,X,Y)
      DIMENSION YY(1),XX(1)
      Y=YY(1)
      IF(NN.EQ.1) RETURN
      DO 1 K=1,NN
      IF(X.LE.XX(K)) GO TO 2
    1 CONTINUE
      Y=YY(NN)
      RETURN
    2 IF(K.EQ.1) RETURN
      Y=((X-XX(K-1))/(XX(K)-XX(K-1)))*(YY(K)-YY(K-1))+YY(K-1)
      RETURN
      END
      SUBROUTINE FOV(ISENS,ISTYPE,FDTS,PFOV)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DELTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      DIMENSION FDTS(20)
      CPI=3.14159
      SIGX=FDTS(5)
      SIGY=FDTS(6)
      IF(ISENS.EQ.1) GO TO 100
      IF(ISTYPE.EQ.2) GO TO 100
      IF(X.LT.X4) GO TO 100
      ANGTC=CPI/2.-ANGLT
      IF(SIGX.GT.0.) GO TO 40
      PFOVX=1.0
      GO TO 50
   40 XW=2.*SR*TAN(THETAH/2.)/(2.828428*SIGX)
      PFOVX=ERRFUN(XW)
   50 IF(SIGY.GT.0.) GO TO 60
      PFOVY=1.0
      GO TO 90
   60 IF((ANGTC+THETAV/2.).LT.(CPI/2.)) GO TO 70
      PY1=.5
      GO TO 80
   70 XL1=H*(TAN(ANGTC+THETAV/2.)-TAN(ANGTC))/(2.828428*SIGY)
      PY1=.5*ERRFUN(XL1)
```

```
   80 XL2=H*(TAN(ANGTC)-TAN(ANGTC-THETAV/2.))/(2.828428*SIGY)
      PY2=.5*ERRFUN(XL2)
      PFØVY=PY1+PY2
   90 PFØV=PFØVX*PFØVY
      RETURN
  100 PFØV=1.0
      RETURN
      END
      FUNCTION ERRFUN(X)
      C=1.12837916709551
      ERRFUN=-1.E50
      IF(X) 6,1,1
    1 X2=X*X
      IF(X2.LT.180.) GØ TØ 10
      DENS=0.
      GØ TØ 11
   10 DENS=C*EXP(-X2)
   11 IF(X-2.25) 4,4,2
C     CØMPUTE ERRFUN USING A CØNTINUED FRACTIØN EXPANSIØN
    2 N=76./X-4.
      A1=X2*(X2+4.5)+2.
      A2=X*(X2*(X2+7.)+8.25)
      B1=X*(X2*(X2+5.)+3.75)
      B2=X2*(X2*(X2+7.5)+11.25)+1.875
      T=3.
      DØ 3 I=3,N
      A3=X*A2+T*A1
      A1=A2
      A2=A3
      B3=X*B2+T*B1
      B1=B2
      B2=B3
    3 T=T+.5
      CERRF=DENS*A3/(2.*B3)
      ERRFUN=1.-CERRF
      RETURN
C     CØMPUTE ERRFUN FRØM THE MXLAURIN PØWER SERIES
    4 N=11.5*X+5.
      T=5.
      ERRFUN=X*(1.-X2*(3.33333333333333E-1-X2*
     $        (.1-X2*(2.38095238095238E-2-X2*4.62962962962963E-3))))
      Y=X*X2*X2*X2*X2/216.
      DØ 5 I=2,N
      Y=-X2*(2.*T-1.)*Y/(T*(2.*T+1.))
      ERRFUN=ERRFUN+Y
    5 T=T+1
      ERRFUN=C*ERRFUN
      CERRF=1.-ERRFUN
    6 RETURN
      END
```

```
      SUBROUTINE ATMOS(FDT,FDB,IDEVT,FDEVT,IDS,FDS,ISENS,
     $TAT,CA,CM)
      COMMON/TABLES/FDTAB(725)
      COMMON/TABSET/ICT(50)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DELTAT
      DIMENSION FDT(20),FDB(20),IDEVT(20),FDEVT(20),
     $IDS(20),FDS(20)
      VG=FDEVT(4)
      IF(ISENS.EQ.2) GO TO 200
      RT=FDT(4)
      RB=FDB(1)
      CI=ABS(RT-RB)/AMAX1(RT,RB)
      TEMP1=.001*H
      TEMP1=.44/(TEMP1+1.2)+.56*EXP(-.08*TEMP1)+.073
      VS=VG/TEMP1
      IF(ISENS.EQ.1) XLAMDA=FDS(4)
      IF(ISENS.EQ.3) XLAMDA=FDS(15)
      IF(ISENS.EQ.4) XLAMDA=FDS(15)
      SIGMA=(3.912/(VS*6076.))*(XLAMDA/.55)
      IF(ISENS.EQ.1) GO TO 5
      GAMMAT=FDS(9)
      IF(ISENS.EQ.3) GO TO 100
    5 TAT=EXP(-SIGMA*SR)
      ITAT=IDEVT(4)
      IF(ITAT.EQ.1) GO TO 300
   10 ICLOUD=IDEVT(1)
      RTE=FDEVT(1)
      TEMP1=.2
      IF(ICLOUD.EQ.2) TEMP1=.6
      IF(ICLOUD.EQ.3) TEMP1=1.
      ZK=TEMP1/RTE
      TEMP1=1.+ZK*(1.-TAT)/TAT
   50 CA=CI/TEMP1
   55 IF(ISENS.EQ.1) RETURN
   60 CM=CA/(2.-CI)
      CM=1.-(1.-CM)**GAMMAT
      RETURN
  100 GAIN=FDS(19)
      ISTYPE=IDS(1)
      IF(ISTYPE.EQ.1) PHID=ANGLT
      TEMP1=PHID-THETAV/2.
      TEMP2=PHID+THETAV/2.
      SRMIN=H/SIN(TEMP1)
      SRMAX=H/SIN(TEMP2)
      ZMIN=2.*SIGMA*SRMIN
      ZMAX=2.*SIGMA*SRMAX
      TAT=EXP(-ZMAX)
      JJ=ICT(1)
      NX=ICT(2)
```

```
      IX=JJ
      IY=IX+NX
      CALL INTER1(FDTAB(IX),FDTAB(IY),NX,ZMIN,F2ZMIN)
      CALL INTER1(FDTAB(IX),FDTAB(IY),NX,ZMAX,F2ZMAX)
      XNA=(GAIN*SIGMA*SRMAX*.25)*
     $     (SRMAX/SRMIN*F2ZMIN-F2ZMAX)
      XNT=XNA+RT*TAT
      XNB=XNA+RP*TAT
      CA=ABS(XNT/XNB-1.)
      GO TO 60
  200 RH=FDEVT(5)
      DFLTB=FDB(2)
      TEMP1=.00013*H
      TEMP1=(1./TEMP1)*(1.-EXP(-TEMP1))
      TEMP2=.00028*H
      TEMP2=(1./TEMP2)*(1.-EXP(-TEMP2))
      XW0=1.432*RH*EXP(.0652*(DFLTB-273.))
      TEMP2=(.278*TEMP2)/(VG*6.076)+.017*XW0*TEMP1
      TAT=EXP(-TEMP2*SR*.001)
      RETURN
  300 IAZIM=IDEVT(5)
      HH=H
      SR1000=.001*SR
      BMAX=AMAX1(RT,RB)
      ZENITH=ANGLT/.01745+90.
      IF(IAZIM.EQ.2) GO TO 350
      JJ=ICT(31)
      NX=ICT(32)
      NY=ICT(33)
      IX=JJ
      IY=IX+NX
      IZ=IY+NY
      CALL INTER2(HH,SR1000,FDTAB(IX),FDTAB(IY),FDTAB(IZ),
     $ NX,NY,DPR,XXXX,NX)
      JJ=ICT(37)
      NX=ICT(38)
      IX=JJ
      IY=IX+NX
      CALL INTER1(FDTAB(IX),FDTAB(IY),NX,ZENITH,RF)
      GO TO 360
  350 JJ=ICT(34)
      NX=ICT(35)
      IX=JJ
      IY=IX+NX
      IZ=IY+NY
      CALL INTER2(HH,SR1000,FDTAB(IX),FDTAB(IY),FDTAB(IZ),
     $ NX,NY,DPR,XXXX,NX)
      JJ=ICT(40)
      NX=ICT(41)
      IX=JJ
```

```
      IY=IX+NX
      CALL INTFR1(FDTAP(IX),FDTAP(IY),NX,ZENITH,RF)
  360 CA=CI/(1.+DPR/(RMAX*RF))
      GO TO 55
      END
      SUBROUTINE VISOR(FDT,FDB,IDEVT,FDEVT,
     $IDS,FDS,P3D,P3R)
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DFLTAT
      COMMON/BLOCK4/X1,X2,X3,X4,X5,X6,XY1,XY2,XY3,XY4,XY5,XY6
      COMMON/BLOCK6/ALPHAH,ALPHAL,ALPHAV
      DIMENSION FDT(20),FDB(20),IDEVT(20),FDEVT(20),
     $IDS(20),FDS(20)
      ISENS=1
      THETA=.8
      CR=.01745
      XK1=.0265
      XK2=.24
      XK3=.44
      XK4=1.6
      CT=XK1*THETA**XK2+(XK3*THETA**XK4)/
     $   ((AMIN1(ALPHAH,ALPHAV)*60./CR)**2)
      CALL ATMOS(FDT,FDB,IDEVT,FDEVT,XXX,FDS,ISENS,
     $XXX,CA,XXX)
      TEMP1=CA/CT
      IF(TEMP1.LT..05) GO TO 90
      IF(TEMP1.LT.1.) GO TO 40
      AK=.43
      GO TO 50
   40 AK=-.57
   50 TEMP1=4.2*((TEMP1-1.)**2)
      P3D=.57+AK*SQRT(1.-EXP(-TEMP1))
      A=3.
      B=2.9E-6
      IF(X.GE.X2) GO TO 55
      VALPHA=0.
      GO TO 60
   55 VALPHA=(SP*SQRT(H*H+Y*Y)/(SR*SR))/CR
   60 TVA=(A+B*(VALPHA**3))/2.
      TEMP1=((AMIN1(ALPHAH,ALPHAV)*60./CR)/TVA)**2
      IF(TEMP1.LE.3.2) GO TO 100
      TEMP1=((TEMP1-3.2)**2)/11.
      IF(TEMP1.LT.180.) GO TO 70
      P3R=1.0
      RETURN
   70 P3R=1.-EXP(-TEMP1)
      RETURN
   90 P3D=0.
  100 P3R=0.
      RETURN
      END
```

```
      SUBROUTINE FLIR(FDT,FDB,FDFVT,IDS,FDS,P3D,P3R)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DFLTAT
      COMMON/BLOCK6/ALPHAH,ALPHAL,ALPHAV
      DIMENSION FDT(20),FDB(20),FDFVT(20),
     $IDS(20),FDS(20)
      ISENS=2
      DELTB=FDB(2)
      EMMB=FDB(3)
      DELTT=FDT(5)
      EMMT=FDT(6)
      G=FDEVT(6)
      ZK1=FDS(5)
      ZK2=FDS(6)
      CALL ATMOS(XXX,FDB,XXX,FDFVT,XXX,XXX,ISENS,
     $TAT,XXX,XXX)
      XNV=AMIN1(ALPHAH,ALPHAL)/AMAX1(ALPHAH,ALPHAL)
      IX=1
  100 TEMP1=0.
      XN=2.*THETAV/AMIN1(ALPHAH,ALPHAV)
      IF(XN.GT.500.) GO TO 320
      IF(IX.NE.2) GO TO 110
      XN=4.*XN
      IF(XN.GT.500.) GO TO 320
      GO TO 130
  110 IF(XN.GT.300.) GO TO 120
      SNRT=6.18*EXP(-.00252*XN)
      GO TO 150
  120 SNRT=2.90176-.002*(XN-300.)
      GO TO 150
  130 IF(XN.GT.300.) GO TO 140
      SNRT=8.2*EXP(-.00248*XN)
      GO TO 150
  140 SNRT=3.89671-.002*(XN-300.)
  150 SNRT=SNRT*(1.+.008*G*G)
      IF(XN.LT.100.) GO TO 160
      SNMRT=3.-.002*XN
      GO TO 170
  160 SNMRT=5.-.022*XN
  170 XMRT=ZK1*EXP(ZK2*XN)
      TDIFF=ABS(DELTT*EMMT-DELTB*EMMB)
      SNRD=(SNMRT*TDIFF*TAT/XMRT)*((DELTB*EMMB/300.)**3)*
     $      SQRT(1./(7.*XNV))
      SNR=SNRD-SNRT
      IF(SNR.LT.5.) GO TO 310
      TEMP1=1.0
      GO TO 320
  310 TEMP1=1.-ERRFUN(ABS(SNR)/1.414214)
      TEMP1=.5*TEMP1
      IF(SNR.GT.0.) TEMP1=1.-TEMP1
```

```
320 IF(IX.EQ.2) GØ TØ 390
330 P3D=TEMP1
    IF(TEMP1.LE.0.) GØ TØ 390
    IX=2
    GØ TØ 100
390 P3R=TEMP1
    RETURN
    END
    SUBRØUTINE TV(ISENS,FDT,FDB,IDEVT,FDEVT,
   $IDS,FDS,IFØV,P3D,P3R)
    CØMMØN/BLØCK1/THETAH,THETAV,PHID
    CØMMØN/BLØCK3/X,Y,XY,H,SR,SP,ANGLØ,ANGLT,DELTAT
    CØMMØN/BLØCK6/ALPHAH,ALPHAL,ALPHAV
    DIMENSIØN FDT(20),FDB(20),IDEVT(20),FDEVT(20),
   $IDS(20),FDS(20)
    ASPECT=FDS(1)
    BAND=FDS(5)
    DIAG=FDS(6)
    FNUM=FDS(6+IFØV)
    GAMMAT=FDS(9)
    XFC=FDS(11)
    XI=FDS(12)
    XIMAX=FDS(13)
    XIP=FDS(14)
    XLAMDA=FDS(15)
    XNL=FDS(16)
    XNR=FDS(17)
    GT=FDS(20)
    RTE=FDEVT(1)
    G=FDEVT(6)
    CALL ATMØS(FDT,FDB,IDEVT,FDEVT,IDS,FDS,ISENS,
   $TAT,XXX,CM)
    EAREA=ASPECT*DIAG*DIAG/(1.+ASPECT*ASPECT)
    RESPØN=10.**((ALØG10(XI)-ALØG10(EAREA))/GAMMAT
   $        -ALØG10(.674*XFC))
    IFLAG=0
    IF(ISENS.EQ.3) GØ TØ 10
    ICLØUD=IDEVT(1)
    ISUNAN=IDEVT(2)
    TRANSM=FDS(10)
    CFAC=1.
    IF(ICLØUD.EQ.2) CFAC=3.1635
    IF(ICLØUD.EQ.3) CFAC=10.
    HS=595.
    IF(ISUNAN.EQ.2) HS=465.
    IF(ISUNAN.EQ.3) HS=265.
    XICØN=RESPØN*HS*RIF*TRANSM*TRANSM/(4.*CFAC)
    GØ TØ 20
 10 PØUT=FDS(18)
    ASTR=THETAV*THETAH
```

```
 20 IF(ISENS.EQ.3) GO TO 30
    XIAVE=EAREA*(XICON/(FNUM*FNUM))*GAMMAT
    GO TO 40
 30 XICON2=RESPON*RTE*POUT*TAT*TRANSM/(4.*ASTR*SR*SR)
    XIAVE=EAREA*(XICON2/(FNUM*FNUM))**GAMMAT
 40 IF(XIAVE.LE.(.85*XIMAX)) GO TO 50
    FNUM=1.414214*FNUM
    IFLAG=1
    GO TO 20
 50 IF(IFLAG.EQ.0) GO TO 60
    PRINT 55,FNUM
 55 FORMAT(* TUBE SATURATION, NEW FNUM= *,F5.2)
 60 XNOISE=XIP*XIP/(2.*BAND)
    XE=SQRT(XNL*XNR/(1.414214*ASPECT))
    XNOE=1.24*XE
    XNET=.31*XE
    XNEL=545.*DIAG/(XLAMDA*FNUM*SQRT(1.+ASPECT*ASPECT))
    XNOL=3.66*XNEL
    XNV=AMIN1(ALPHAH,ALPHAL)/AMAX1(ALPHAH,ALPHAL)
    IX=1
100 TEMP1=0.
    XN=2.*THETAV/AMIN1(ALPHAH,ALPHAV)
    IF(XN.GT.700.) GO TO 320
    IF(IX.NE.2) GO TO 110
    XN=4.*XN
    IF(XN.GT.700.) GO TO 320
    GO TO 130
110 IF(XN.GT.300.) GO TO 120
    SNRT=6.18*EXP(-.00252*XN)
    GO TO 150
120 SNRT=2.90176-.002*(XN-300.)
    GO TO 150
130 IF(XN.GT.300.) GO TO 140
    SNRT=8.2*EXP(-.00248*XN)
    GO TO 150
140 SNRT=3.89671-.002*(XN-300.)
150 SNRT=SNRT*(1.+.008*G*G)
    ZNET=(XN/(XNV+XNET))**2
    ZNEL=(XN/(XNV+XNEL))**2
    PSIY=SQRT(1.+ZNET*ZNEL)
    GAMMAY=PSIY/SQRT(1.+ZNEL+2.*ZNET)
    IF(XN.GT.(XNOE/3.)) GO TO 210
    BETAT=(.3*XE/XN)*ERRFUN(3.*XN/XE)
    GO TO 220
210 BETAT=(XE/XN)*(.086+.233*ERRFUN(2.45*XN/XE))
220 IF(XN.LE.10.) GO TO 240
    IF(XN.GT.XNOE) GO TO 250
    I=-1
    RSOF=0.
225 I=I+2
```

```
      IF(I.GT.50) GO TO 260
      XIN=I*XN
      IF(XIN.GT.XNOE) GO TO 260
      PHIX=XIN/XNOL
      RLN=.637*(ACOS(PHIX)-PHIX*SQRT(1.-PHIX*PHIX))
      IF(XIN.GT.(XNOE/3.)) GO TO 230
      RSOF=RSOF+(.811*RLN*EXP(-4.4*XIN*XIN/(XE*XE)))/(I*I)
      GO TO 225
  230 RSOF=RSOF+(.637*RLN*EXP(-3.*XIN*XIN/(XE*XE)))/(I*I)
      GO TO 225
  240 RSOF=1.0
      GO TO 260
  250 RSOF=0.
      GO TO 320
  260 SNRD=(2.*RSOF*CM*GT*XIAVE/XN)*SQRT(.1*XNV/(ASPECT*
     $     PSIY*(GT*GT*BETAT*GAMMAY*XIAVE*1.6E-19+XNOISE)))
      SNR=SNRD-SNRT
      IF(SNR.LT.5.) GO TO 310
      TEMP1=1.0
      GO TO 320
  310 TEMP1=1.-ERRFUN(ABS(SNR)/1.414214)
      TEMP1=.5*TEMP1
      IF(SNR.GT.0.) TEMP1=1.-TEMP1
  320 IF(IX.EQ.2) GO TO 350
      IF(RSOF.GT.0.) GO TO 330
      P3D=0.
      P3R=0.
      RETURN
  330 P3D=TEMP1
      IX=2
      GO TO 100
  350 IF(RSOF.GT.0.) GO TO 390
      P3R=0.
      RETURN
  390 P3R=TEMP1
      RETURN
      END
      SUBROUTINE SENFLR(ISENS,IDT,FDT,FDB,FDEVT,
     $ FDTS,IDS,FDS,P2,P3D,P3R)
      COMMON/TABLES/FDTAB(725)
      COMMON/TABSET/ICT(50)
      COMMON/BLOCK1/THETAH,THETAV,PHID
      COMMON/BLOCK3/X,Y,XY,H,SR,SP,ANGLO,ANGLT,DELTAT
      DIMENSION IDT(20),FDT(20),FDB(20),FDEVT(20),
     $ FDTS(20),IDS(20),FDS(25)
C
C
      CPI=3.14159265
      C2=.3048
      C3=1.689
```

```
      CR=.01745
      SNDB=0.
      VC=0.
      NT=IDT(1)
      PN=0.
      VRT=0.
      VB=0.
C
      G=FDEVT(6)
      VTT=FDT(12)
      C= 3.E 8
      H=H*C2
      VG=SP/1.687778
      GO=FDS(2)
      PRF=FDS(4)
      PX=FDS(5)
      FLAMDA=FDS(6)
      WS=FDS(7)
      PHIM=PHID-THETAV/2.
      VP=FDEVT(7)
      TAUPW=FDS(12)
      TAUP1=FDS(13)
      TAUP2=FDS(14)
      VF=FDS(15)
      VPRF=FDS(16)
      CAS=FDS(12)
      ISEAS=IDS(1)
      DTX=FDT(1)*C2
      DTY=FDT(2)*C2
      PSI=FDT(11)*C8
      XXXX=0.
      WP=FDE(5)
      GG = 0.
      PNR=FDS(3)
C SELECT BAND
      IBAND=1
      IF(FLAMDA.LT.(.025))IBAND=2
      IF(FLAMDA.LT.(.0125)) IBAND=3
      BAND=IBAND
C COVERAGE
      Y=Y*C2
C
      P3D=0.
      X=X*C2
C
C GEOMETRY
      SR=SR*C2
      PHI=ANGLT
      DGY=SR*THETAH
      DGX=C*TAUPW/(2.*COS(PHI))
```

```
      TSPACE=FDT(10)*C2
      RNJC=NT
      DGMIN=AMIN1(DGX,DGY)
      RNJC1=DGMIN/(NT*(TSPACE+DTX))
      RATSIG=AMIN1(RNJC,RNJC1)
      SIGFAC=AMAX1(1.,RATSIG)
      XTE=AMIN1(DTX,DGX)
      YTE=AMIN1(DTY,DGY)
      AG=DGX*DGY
      ATE=XTE*YTE
      ABE=AMAX1((AG-ATE),0.)
      TEMP1=COS(PHI)**2
      TEMP1=SQRT(SR*SR*TEMP1-Y*Y)
      ALPHA=ATAN2(Y,TEMP1)
      VRT=VTT*COS(PHI)*COS(PSI-ALPHA)
      TEMP1=SIN(PHI)**2
      TEMP2=SIN(PHIM)**2
      TEMP3=SQRT(COS(PHI))
      TEMP4=SQRT(COS(PHIM))
      GAF=GO*TEMP2*TEMP3/(TEMP1*TEMP4)
C RESOLUTION GEOMETRY
      AG1=AG
      ABE1=0.
C
C
      IF(ABE.EQ.0.)GO TO 200
  100 DX=AMAX1(DGX,DGY)
      DN=AMIN1(DGX,DGY)
      IF(DX.LE.WP) GO TO 200
      IF(DN.LE.WP) GO TO 110
      AP=WP*(DX+DN)/2.
      GO TO 120
  110 AP=DN*(DX+WP)/2.
  120 AL=AMAX1((AG-AP),0.)
      F1=1.0
      TEMP1=F1
      F1=F1/TEMP1
      A1=AP+F1*AL
      IF(ATE.GT.AP) GO TO 130
      ABE1=A1-ATE
      GO  TO  140
  130 ATEL=ATE-AP
      ABE1=A1-F1*ATEL-AP
  140 AG1=F1*AG
C RADAR CROSS SECTION
  200 TEMP1=SIN(PHI)
      AT=DTX*DTY
      I=6+IBAND
      SIGTE=FDT(I)*(.3/FLAMDA)**.3
      SIGTE=10.*ALOG10(SIGTE)+PHI/(CR*15.)
```

```
      TEMP5=PSI
      IF(PSI.GT..7854) TEMP5=CPI/2.-PSI
      SIGTE=SIGTE-5/45.*PSI/CB
      SIGTE=10.**(SIGTE/10.)
      SIGTF=SIGFAC*SIGTE
  210 TEMP2=AG/AT
      IF(TEMP2.GE.1.) GO TO 220
      SIGTE=SIGTE*TEMP2
C
  220 PHI=PHI/CB
      JJ=ICT(10)
      NX=ICT(11)
      NY=ICT(12)
      IX=JJ
      IY=IX+NX
      IZ=IY+NY
      CALL INTER2(PHI,BAND,FDTAB(IX),FDTAB(IY),FDTAB(IZ),
     $ NX,NY,SIGB1,XXXX,NX)
      SIGB1=10.**(SIGB1/10.)
  222 PHI=PHI*CB
      SIGBF=TEMP1*(ABE1*SIGB1)
      SIGG=TEMP1*(AG1*SIGB1)
C
      CALL ATTEN(IBAND,CAS,H,ISEAS,PHI,TAUAR)
      TAUAR=1./(10.**(TAUAR/10.))
  310 PNB=0.
C
C RECEIVER POWER
C
      TEMP3=4*CPI*CPI*SR*SR
      PD0=TAUP1*PX*GAF*TAUAR*CPI/TEMP3
      PRTT=SIGTF*PD0
      PRBT=SIGBF*PD0
      PRRB=SIGG*PD0
      TEMP1=TAUP2*GAF*TAUAR*FLAMDA**2/(4.*TEMP3)
      PATT=TEMP1*PRTT
      PABT=TEMP1*PRBT
      PARB=TEMP1*PRRB
      TEMP1=THETAH*PRF/(WS*CB)
      TEMP1=SQRT(TEMP1)
      PTT=TEMP1*PATT
      PTB=TEMP1*PABT
      PRB=TEMP1*PARB
      SNRAT=(PTT+PTB)/(PRB+PNR)
      IF(ISENS.EQ.6) GO TO 361
C
C CLUTTER FREQUENCY
C
      VR=0.
  330 TEMP1=SIN(PHI)
```

```
      TEMP2=COS(PHI)
      TEMP3=SIN(ALPHA)
      TEMP4=COS(ALPHA)
      VVY=C2*C3*VG*TAUPW*TEMP4*TEMP1**3*C/(TEMP2*FLAMDA*H)
      VVX=C2*C3*VG*THETAH*TEMP3/FLAMDC
      VW=2.*VW/FLAMDA
      VS=FDS(7)*C8/(SQRT(2.*CPI)*THETAH)
      VC=SQRT(VVY**2+VVX**2+VW**2+VR**2+VS**2)
C
C CLUTTER ATTENUATION
C
      VT=SQRT(VC**2+VF**2+VPRF**2)
      VTH=FDS(23)
      VB=FLAMDA*PRF/(2.*C3*C2)
      IF((2.*VTH).LT.VB) GO TO 333
      PRINT 332,X,Y
  332 FORMAT(1X,35HTHRESHOLD SPEED IS GREATER THAN 1/2,
     $12H BLIND SPEED,5X,2HX=E12.5,2X,2HY=E12.5)
      P3D=0.
      GO TO 370
  333 FN1=(VRT-VTH)/VB
      N1=FN1+.00001
      IF(FN1.LT.0.) N1=N1-1
      N1=N1+1
      FN1=N1
      TEMP1=(VRT+VTH)/VB
      IF(FN1.GE.TEMP1) GO TO 334
      P3D=0.
      GO TO 370
  334 FTH=2.*C2*C3*VTH/FLAMDA
      FMF = FDS(1)
      IF (FMF) 9,7,9
    7 FMF = 12.
      PRINT 8
    8 FORMAT(5X,42HAN MF VALUE WAS NOT INPUT SO MF IS SET=12.)
    9 FII = FMF/6.
      FKK = 1./(1.414 *FTH**FII)
      FRT = 2.*C2 *C3 *VRT /FLAMDA
      FBK = FTH *1.414 **(1./FII)
      GG = FKK **2 *FRT **(2.*FII)
      IF (FRT .GT. FBK) GG = 1.
      PTT1 = PTT *GG
      TEMP1=0.
      CALL GAIN(TEMP1,FBK,TEMP2,VT,0,XXXX)
      TEMP2=.5-(TEMP2/(VT*SQRT(2.*CPI)))
      IF (TEMP2 .LT. 0.) TEMP2 = 0.
      PN=2.*TEMP2
      TEMP1 = 0.
      CALL GAIN(TEMP1,FBK,TEMP2,VT,-1,FII)
      PN = PN +2.*FKK **2 /(SQRT(2.*CPI)*VT)*TEMP2
```

```
      PRC=PN*(PRB+PVR)
      PTC=PN*(PTB+PVR)
      SNRAT=(PTT1+PTC)/(PVR+PRC)
  361 SNDB=10.*ALOG10(SNRAT)
  365 TSN=FDS(21)
      P3D=0.
      P3R=0.
      IF(SNDB.GE.TSN)P3D=1.
      IF(P3D.LE.0.) GO TO 400
  370 CONTINUE
      XNR=AMIN1(DTX,DTY)/DGX
      TEMP1=(XNR-3.2)**2/11.
      IF(TEMP1.LT.180.) GO TO 390
      P3R=1.0
      GO TO 400
  390 P3R=1.-EXP(-TEMP1)
  400 AT=DGX*DGY
      AT=AT*(NT+(NT-1)*TSPACE/DGX)
      DSX=FDTS(2)
      DSY=FDTS(3)
      XMAX=H/TAN(PHID-THETAV/2.)
      XMIN=H/TAN(PHID+THETAV/2.)
      YMAX=2.*XMAX*TAN(THETAH/2.)
      YMIN=2.*XMIN*TAN(THETAH/2.)
      IF((DSX.GT.0.).OR.(DSY.GT.0.)) GO TO 450
      AS=(XMAX-XMIN)*((YMAX+YMIN)/2.)
      GO TO 500
  450 DDX=XMAX-XMIN
      IF(DSX.GT.0.) DDX=AMIN1(DSX,DDX)
      DDY=(YMAX+YMIN)/4.-Y
      IF(DSY.GT.0.) DDY=AMIN1((DSY/2.),DDY)
      IF((DDY+Y).GT.(YMIN/2.)) GO TO 470
      AS=2.*DDY*DDX
      GO TO 500
  470 AA=((DDY+Y)-(YMIN/2.))/TAN(THETAH/2.)
      BB=AA*TAN(THETAH/2.)
      AS=DDY*DDX-.5*AA*BB
      DDY=(YMAX+YMIN)/4.+Y
      IF(DSY.GT.0.) DDY=AMIN1((DSY/2.),DDY)
      IF((DDY-Y).GT.(YMIN/2.)) GO TO 490
      AS=AS+DSY*DSX
      GO TO 500
  490 AA=((DDY-Y)-(YMIN/2.))/TAN(THETAH/2.)
      BB=AA*TAN(THETAH/2.)
      AS=AS+DDY*DDX-.5*AA*BB
  500 ARATIO=AT/AS
      TEMP1=(700./G)*ARATIO*DELTAT
      IF(TEMP1.LT.180.) GO TO 510
      P2=1.0
      GO TO 520
```

```
  510 P2=1.-EXP(-TEMP1)
  520 X=X/C2
      Y=Y/C2
      H=H/C2
      SR=SR/C2
      RETURN
      END
      SUBROUTINE INTER2(TARGX,TARGY,X,Y,TAB,NX,NY,ANS,IFLAG,N)
C
C  PERFORMS BIVARIATE,INTERPOLATION
C
      DIMENSION TAB(N,1),X(1),Y(1)
      IFLAG=0
      IF(TARGX-X(1)) 2,3,3
    2 IFLAG=-1
      ARGX=X(1)
      GO TO 4
    3 ARGX=TARGX
    4 IF(TARGY-Y(1)) 5,6,6
    5 IFLAG=-1
      ARGY=Y(1)
      GO TO 7
    6 ARGY=TARGY
    7 DO 8 II=2,NX
      IF(ARGX-XX(II)) 9,9,8
    8 CONTINUE
      II=NX
      IFLAG=1
      ARGX=X(II)
    9 I=II
      DO 10 JJ=2,NY
      IF(ARGY-Y(JJ)) 11,11,10
   10 CONTINUE
      JJ=NY
      IFLAG=1
      ARGY=Y(JJ)
   11 J=JJ
      DY=Y(J)-Y(J-1)
      CY1=-(ARGY-Y(J))/DY
      CY2=(ARGY-Y(J-1))/DY
      DX=X(I)-X(I-1)
      CX1=-(ARGX-X(I))/DX
      CX2= (ARGX-X(I-1))/DX
      ANS=CY1*(CX1*TAB(I-1,J-1)+CX2*TAB(I,J-1))
     1    +CY2*(CX1*TAB(I-1,J)+CX2*TAB(I,J))
      RETURN
      END
      SUBROUTINE GAIN(X1,X2,Y,VT,ITYPE,FI)
      DX=(X2-X1)/100.
      Y=0.
```

```
      X=X1
      DØ 40 I=1,101
      IF(ITYPE) 5,5,6
    5 TEMP=-X*X*.5/(VT*VT)
      TEMP=EXP(TEMP)
      IF(ITYPE.LT.0) TEMP=TEMP*X**(2.*FI)
      GØ TØ 7.
    6 TEMP=SQRT(CØS(X))/(SIN(X)**2)
    7 IF(I.EQ.1) GØ TØ 30
      IF(I.EQ.101) GØ TØ 30
      IF((I/1)*2-I) 10,20,30
   10 TEMP=2.*TEMP
      GØ TØ 30
   20 TEMP=4.*TEMP
   30 Y=Y+TEMP
   40 X=X+DX
      Y=DX*Y/3.
      RETURN
      END
      SUBRØUTINE SAR(FDØ,IDT,FDT,FDB,FDEVT,
     &IDS,FDS,P2,P3D,P3R)
      CØMMØN/TABLES/FDTAB(725)
      CØMMØN/TARSET/ICT(50)
      CØMMØN/BLØCK3/X,Y,XY,H,SR,SP,ANGLØ,ANGLT,DELTAT
      DIMENSIØN FDØ(20),IDT(20),FDT(20),FDB(20),
     &FDEVT(20),IDS(20),FDS(25)
      CAS=FDS(19)
      ISEAS=IDS(1)
      IBAND=1
      FKT=1.
      CPI=2*ASIN(1.)
      C2=.3048
      C3=1.689
      C8=.01745
      C=3.E8
      DX=FDS(1)
      TAUPW=FDS(13)
      FLAMDA=FDS(9)
      IF(FLAMDA.LT..025) IBAND =2
      IF(FLAMDA.LT..0125) IBAND =3
      XXXX=0.
      DTX=FDT(1)*C2
      DTY=FDT(2)*C2
      GZER=FDS(2)
      THETAV=FDS(12)*C8
      SL1=FDS(20)
      SL2=FDS(21)
      SL3=FDS(22)
      PX=FDS(5)
      TAUP1=FDS(14)
```

```
      TAUP2=FDS(15)
      TSN=FDS(7)
      PNR=FDS(3)
      PRF=FDS(4)
      PSI=FDT(11)*C8
      RC=FDS(6)
      Z=6367650.
      AH=FDS(18)*C8
      VG1=FD0(5)
      H=FD0(2)*C2
      NT=IDT(1)
      RNJC=NT
      TSPACE=FDT(10)*C2
      Y=FD0(1)*C2
      WS=FDS(23)*6076.*C2
      G=FDEVT(6)
      IF(Y.LT.(1.5*WS)) GO TO 20
      GO TO 30
   20 Y=1.5*WS
      PRINT 25
   25 FORMAT(* Y TOO SMALL, INCREASED TO BE 1.5 SWATH WIDTHS *)
   30 GAMMA=Y/Z
  222 SR=SQRT(2*Z*(Z+H)*(1-COS(GAMMA))+H*H)
      GRA=SR*TAN(AH)
      SR=SQRT(SR*SR+GRA*GRA)
      HORIZ=SQRT((Z+H)**2-Z*Z)
      PHIH=ASIN(HORIZ/(Z+H))
      PHI=ACOS(((Z+H)**2-SR**2-Z*Z)/(-2*Z*SR))
      PHI=PHI-CPI/2.
      FLP=C*TAUPW*.5/RC
      DGY=FLP/(COS(PHI)*COS(AH))
      DGX=DX/COS(AH)
      DGMIN=AMIN1(DGX,DGY)
      RNJC1=DGMIN/(NT*(TSPACE+DTX))
      RATSIG=AMIN1(RNJC,RNJC1)
      SIGFAC=AMAX1(1.,RATSIG)
      XTE=AMIN1(DGX,DTX)
      XTY=AMIN1(DGY,DTY)
      ATE=XTE*XTY
      AG=DGX*DGY
      ABE=AMAX1((AG-ATE),0.)
      PHIM=PHI-THETAV
      IF(PHIM.LE.PHIH) PHIM=PHIH
      COMP=CPI/2.-PHIM
      IF(PHIM.LE.PHIH)  GO TO 51
      COMP=CPI-ASIN(SIN(CPI/2.-PHIM)*(Z+H)/Z)
   51 GAMMAM=CPI/2.+PHIM-COMP
      YMAX=Z*PHIH
      GAMAXY=ACOS((SR**2-(Z+H)**2-Z*Z)/(-2*Z*(Z+H)))
      XY=Z*GAMAXY/C2
```

```
      PHIM1=PHI-(Y+.5*WS)/Z
      GAMMAX=ACOS((GPA**2-2*Z*Z)/(-2*Z*Z))
      X=Z*GAMMAX/C2
      GAF=GZER*(SIN(PHIM1)/SIN(PHI))**2*SQRT
     &(COS(PHI)/COS(PHIM1))/COS(AH)
      WP=FDR(5)
      AG1=AG
      ABE1=0.
      IF(ABE.EQ.0.) GO TO 200
  105 DXX=AMAX1(DGX,DGY)
      DN=AMIN1(DGX,DGY)
      IF(DXX.LE.WP) GO TO 200
      IF(DN.LE.WP) GO TO 110
      AP=WP*(DXX+DN)/2.
      GO TO 120
  110 AP=DN*(DXX+WP)/2.
  120 AL=AMAX1((AG-AP),0.)
      F1=1.
      A1=AP+F1*AL
      IF(ATE.GT.AP) GO TO 130
      ABE1=ATE-AP
      GO TO 140
  130 ATEL=ATE-AP
      ABE1=A1-F1*ATEL-AP
  140 AG1=F1*AG
  200 AT=DTX*DTY
      PHI=PHI/C8
      N=15
      I=6+IBAND
      BAND=IBAND
      SIGTE=FDT(I)*(.3/FLAMDA)**.3
      TEMP5=ABS(PSI-AH)
      SIGTE=10.*ALOG10(SIGTE)+PHI/15.-5/45.*TEMP5/C8
      IF(TEMP5.GE..7854) SIGTE=SIGTE+5/45.*TEMP5/C8
     &-5/45.*(90.-TEMP5/C8)
      SIGTE=10.**(SIGTE/10.)
  210 TEMP3=AG/AT
      IF(TEMP3.GT.1.) GO TO 220
      SIGTE=TEMP3*SIGTE
  220 SIGTE=SIGFAC*SIGTE
      SIGB1=0.
      JJ=ICT(10)
      NX=ICT(11)
      NY=ICT(12)
      IX=JJ
      IY=IX+NX
      IZ=IY+NY
      CALL INTER2(PHI,BAND,FDTAB(IX),FDTAB(IY),FDTAB(IZ),
     $ NX,NY,SIGB1,XXXX,NX)
      SIGB1=10.**(SIGB1/10.)
```

```
      PHI=PHI*CR
      SIGBE=ABE1*SIGB1*SIN(PHI)
      SIGG=AG1*SIGB1*SIN(PHI)
      CALL ATTEN(IBAND,CAS,H,ISEAS,PHI,TAUAR)
      TAUAR=1./(10.**(TAUAR/10.))
      Q=0.
      TEMP3=0.
      IF(SL1) 270,280,270
  270 TEMP3=1./(10.**(SL1/10.))
  280 IF(SL2) 290,300,290
  290 TEMP3=TEMP3+1./(10.**(SL2/10.))
  300 IF(SL3) 310,320,310
  310 TEMP3=TEMP3+1./(10.**(SL3/10.))
  320 Q=2.*TEMP3
      IF(Q.EQ.0.) Q=.1
      PDO=TAUP1*PX*GAE/(4.*CPI*SR*SR)*TAUAR
      PRTT=TAUAR*SIGTE*PDO
      PRBT=TAUAR*SIGBE*PDO
      PRRB=TAUAR*SIGG*PDO
      TEMP4=TAUP2*TAUAR*GAE*FLAMDA**2/(4*CPI*SR)**
      PATT=PRTT*TEMP4
      PABT=PRBT*TEMP4
      PARB=PRRB*TEMP4
      FL=.44*FLAMDA*SR/DGX
      FN=FL*PF  (C2*C3*VG1)
      PBT=FN*PABT
      PTT=FN*PATT
      PRB=FN*PARB
      PSL=Q*PRB
      SNRAT=(PTT+PBT)/(PRB+PSL+PNR)
      SNRAT=10.*ALOG10(SNRAT)
      Y=Y/C2
      ANGLO=ATAN(Y/X)
      SR=SR/C2
      H=H/C2
      SP=VG1*C3
      ANGLT=PHI
      P3D=0.
      P3R=0.
      IF(SNRAT.GE.TSN) P3D=1.0
      IF(P3D.LE.0.) GO TO 400
      XNR=AMIN1(DGX,DGY)/SQRT(DGX*DGY)
      TEMP1=(XNR-3.2)**2/11.
      IF(TEMP1.LT.180.) GO TO 390
      P3R=1.0
      GO TO 400
  390 P3R=1.-EXP(-TEMP1)
  400 AT=DGX*DGY
      AT=AT*(NT+(NT-1)*TSPACE/DGX)
      ASPECT=FDS(17)
```

```
      WSM=VS
      AS=VSM*WSM*ASPECT
  500 ARATIO=AT/AS
      TVIEW=ASPECT*WSM/(SP*C2)
      TEMP1=(700./G)*ARATIO*TVIEW
      IF(TEMP1.LT.180.) GO TO 510
      P2=1.0
      RETURN
  510 P2=1.-EXP(-TEMP1)
      RETURN
      END
      SUBROUTINE ATTEN(IBAND,CAS,H,ISFAS,PHI,ATT)
      COMMON/TABLES/FDTAB(725)
      COMMON/TABSET/ICT(50)
      JJ1=ICT(13)
      JJ2=ICT(16)
      JJ3=ICT(19)
      JJ4=ICT(22)
      JJ5=ICT(25)
      JJ6=ICT(28)
      NX1=ICT(14)
      NY1=ICT(15)
      NX2=ICT(17)
      NY2=ICT(18)
      NX3=ICT(20)
      NY3=ICT(21)
      NX4=ICT(23)
      NY4=ICT(24)
      NX5=ICT(26)
      NY5=ICT(27)
      NX6=ICT(29)
      NY6=ICT(30)
      IZ=JJ2+NX2+NY2
      I1=IZ+NY2+ISFAS
      I2=IZ+ISFAS
      CPI=2.*ASIN(1.)
      IF(H.LT.FDTAB(I1)) GO TO 50
      IF(H.LT.FDTAB(I2)) GO TO 70
      DELS=SI(FDTAB(I1),H,PHI)-SI(FDTAB(I2),H,PHI)
      GO TO 100
   50 DELS=0.
      GO TO 100
   70 DELS=SI(FDTAB(I1),H,PHI)
  100 SUM=0.
      IZ=JJ4+NX4+NY4
      DO 200 I=1,NY4
      N=IZ+NY4*(ISFAS-1)+I
      SUM=SUM+FDTAB(N)
      IF(SUM.GE.CAS) GO TO 250
  200 CONTINUE
```

```
250 C=.06
    IF(IBAND.EQ.2) C=.2
    IF(IBAND.EQ.3) C=.4
    II=JJ3+NY3+NX3+NY3*(ISEAS-1)+I
    ATT=C*FDTAB(II)*.001*DFLS*(I-1)/8
    IF(H.LT.FDTAB(I2)) GO TO 260
    DFLS=S1(0,H,PHI)-S1(FDTAB(I1),H,PHI)
    GO TO 280
260 DFLS=S1(0,H,PHI)
280 SUM=0.
    MC=0
    IF(DFLS.LT.10000.) MC=1
    IZ=JJ5+NX5+NY5
    IF(MC.EQ.1) IZ=JJ6+NX6+NY6
    IF(MC.EQ.1) NY5=NY6
    IF(MC.EQ.1) NX5=NX6
    DO 300 I=1,NY5
    N=IZ+NY5*(ISEAS-1)+I
    SUM=SUM+FDTAB(N)
    IF(SUM.GE.CAS) GO TO 350
300 CONTINUE
350 IZ=JJ1+NX1+NY1
    ILOC=IZ+NY1*(IBAND-1)+I
    ATR=FDTAB(ILOC)*DFLS
    ATT=ATT+ATR
    RETURN
    END
    FUNCTION S1(H1,H,PHI)
    CPI=2.*ASIN(1.)
    Z=6367650.
    S1=-(Z+H)*COS(CPI/2.-PHI)+SQRT(((Z+H)*COS(CPI/2.-PHI))**2
   &+(Z+H)**2-(Z+H1)**2)
    RETURN
    END
```

DAT
FILM
6–8